

# AM with Multiple Merlins

Scott Aaronson\*

Russell Impagliazzo†

Dana Moshkovitz‡

## Abstract

We introduce and study a new model of interactive proofs:  $\text{AM}(k)$ , or Arthur-Merlin with  $k$  non-communicating Merlins. Unlike with the better-known  $\text{MIP}$ , here the assumption is that each Merlin receives an *independent* random challenge from Arthur. One motivation for this model (which we explore in detail) comes from the close analogies between it and the quantum complexity class  $\text{QMA}(k)$ , but the  $\text{AM}(k)$  model is also natural in its own right.

We illustrate the power of multiple Merlins by giving an  $\text{AM}(2)$  protocol for 3SAT, in which the Merlins' challenges and responses consist of only  $n^{1/2+o(1)}$  bits each. Our protocol has the consequence that, assuming the Exponential Time Hypothesis (ETH), any algorithm for approximating a dense CSP with a polynomial-size alphabet must take  $n^{(\log n)^{1-o(1)}}$  time. Algorithms nearly matching this lower bound are known, but their running times had never been previously explained. Brandao and Harrow have also recently used our 3SAT protocol to show quasipolynomial hardness for approximating the values of certain entangled games.

In the other direction, we give a simple quasipolynomial-time approximation algorithm for free games, and use it to prove that, assuming the ETH, our 3SAT protocol is essentially optimal. More generally, we show that multiple Merlins never provide more than a polynomial advantage over one: that is,  $\text{AM}(k) = \text{AM}$  for all  $k = \text{poly}(n)$ . The key to this result is a *subsampling theorem for free games*, which follows from powerful results by Alon et al. and Barak et al. on subsampling dense CSPs, and which says that the value of any free game can be closely approximated by the value of a logarithmic-sized random subgame.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Restricting to Independent Questions . . . . .	3
<b>2</b>	<b>Our Results</b>	<b>4</b>
2.1	Upper Bounds . . . . .	4
2.2	Hardness Results . . . . .	5
<b>3</b>	<b>Detailed Overview of Results</b>	<b>6</b>
3.1	3SAT Protocol . . . . .	6
3.2	Approximation Algorithms for Free Games . . . . .	8
3.3	Generalizing to $k$ Merlins . . . . .	10

---

\*MIT. Email: aaronson@csail.mit.edu. Supported by the National Science Foundation under Grant No. 0844626, a TIBCO Chair, and an Alan T. Waterman award.

†UCSD. Email: russell@cs.ucsd.edu. Supported by the Simons Foundation, the Ellentuck Fund, the Friends of the Institute for Advanced Study, and NSF grants DMS-0835373, CCF-121351, and CCF-0832797 subcontract no. 00001583.

‡MIT. Email: dmoshkov@mit.edu. Supported by the National Science Foundation under Grant No. 1218547.

<b>4</b>	<b>Quantum Motivation</b>	<b>11</b>
4.1	Connection to Our Results . . . . .	15
<b>5</b>	<b>Preliminaries</b>	<b>15</b>
<b>6</b>	<b>Analysis of the Birthday Game</b>	<b>18</b>
6.1	The Basic Result . . . . .	18
6.2	The High-Error Case . . . . .	22
6.3	The Low-Error Case . . . . .	25
6.4	Complexity Consequences . . . . .	28
<b>7</b>	<b>Limitations of Multi-Prover AM</b>	<b>28</b>
7.1	The Basic Approximation Algorithm . . . . .	28
7.2	Subsampling for Free Games and $\text{AM}(2) = \text{AM}$ . . . . .	33
7.3	The $k$ -Merlin Case . . . . .	36
7.4	Subsampling with $k$ Merlins . . . . .	40
<b>8</b>	<b>Conclusions and Open Problems</b>	<b>44</b>
<b>9</b>	<b>Acknowledgments</b>	<b>46</b>

# 1 Introduction

The PCP characterization of NP [6, 7], with the resulting hardness of approximation results, is one of the great achievements of computational complexity. Leading up to this work was another landmark result, the 1991 theorem of Babai, Fortnow, and Lund [8] that  $\text{MIP} = \text{NEXP}$ , where  $\text{MIP}$  is Multi-Prover Interactive Proofs and  $\text{NEXP}$  is Nondeterministic Exponential Time. Both of these results can be paraphrased as characterizing the hardness of a certain computational problem from game theory: estimating the value of a two-player cooperative game with simultaneous moves. Such games are known in the complexity community as *two-prover games*, and in the quantum information community as *nonlocal games*. From now on, we will use the term two-prover games.

**Definition 1 (Two-Prover Games)** *A two-prover game  $G$  consists of:*

- (1) *finite question sets  $X, Y$  and answer sets  $A, B$ ,*
- (2) *a probability distribution  $\mathcal{D}$  over question pairs  $(x, y) \in X \times Y$ , and*
- (3) *a verification function  $V : X \times Y \times A \times B \rightarrow [0, 1]$ .<sup>1</sup>*

*The value of the game, denoted  $\omega(G)$ , is the maximum, over all pairs of response functions  $a : X \rightarrow A$  and  $b : Y \rightarrow B$ , of*

$$\mathbb{E}_{(x,y) \sim \mathcal{D}} [V(x, y, a(x), b(y))]. \quad (1)$$

---

<sup>1</sup>In most of the actual games we will consider,  $V$  will take values in  $\{0, 1\}$  only. However, the possibility of real  $V$  is needed for full generality.

The interpretation is this: the game  $G$  involves a verifier/referee Arthur, as well as two cooperating provers Merlin<sub>1</sub> and Merlin<sub>2</sub>, who can agree on a strategy in advance but cannot communicate once the game starts. First Arthur chooses a pair of questions  $(x, y)$  from  $\mathcal{D}$ , and sends  $x$  to Merlin<sub>1</sub> and  $y$  to Merlin<sub>2</sub>. The Merlins then send back responses  $a = a(x)$  and  $b = b(y)$  respectively.<sup>2</sup> Finally, Arthur declares the Merlins to have “won” with probability equal to  $V(x, y, a, b)$ . Then  $\omega(G)$  is just the probability that the Merlins win if they use an optimal strategy.

It is not hard to show that computing the exact value of a two-prover game is NP-hard. The PCP Theorem can be interpreted as saying that even to *approximate* the value to within an additive constant is also NP-hard. To make this precise, we can define the classes PCP and MIP as those decision problems *polynomial-time reducible* to approximating the value of a two-prover game. The difference between the classes is that for PCP’s, the reduction computes an *explicit* description of the game, whereas for MIP, the description is *implicit*.

To be more precise, we start with a decision problem  $L$ . Given an instance  $I$  of  $L$ , a reduction constructs a two-prover game  $G_I$  with the following properties:

- **(Completeness)** If  $I \in L$  then  $\omega(G_I) \geq 2/3$ .
- **(Soundness)** If  $I \notin L$  then  $\omega(G_I) \leq 1/3$ .
- **(Efficiency)** In the “explicit” case, the sets  $X, Y, A, B$  can be generated in time polynomial in  $n = |I|$ , the distribution  $\mathcal{D}$  can be described in polynomial time as the uniform distribution over some subset of  $X \times Y$ , and the verification procedure  $V(x, y, a, b)$  can be generated in polynomial time as a table of size  $|X| \times |Y| \times |A| \times |B|$ . In the “implicit” case,  $X, Y, A, B$  are sets of poly( $n$ )-bit strings,  $\mathcal{D}$  can be described as a probabilistic polynomial-time sampling procedure that returns a pair  $(x, y) \in X \times Y$ , and the verification function  $V(x, y, a, b)$  can be computed in polynomial time.

The class PCP then consists of all decision problems that can be reduced explicitly to two-prover games, while MIP consists of all decision problems that can be reduced implicitly to two-prover games. As frequently happens, switching from explicit to implicit representations causes us to “jump up” in complexity by an exponential. The dual theorems PCP = NP and MIP = NEXP bear out this general pattern.

The hardness of approximating two-prover games can in turn be used to show hardness of approximation for many constraint-satisfaction problems. Better trade-offs in the parameters of the reduction and specific kinds of verification procedure give tighter hardness of approximation results for a wide variety of particular combinatorial optimization problems. So the study of two-prover games did not end with the PCP Theorem.

## 1.1 Restricting to Independent Questions

In this paper, we consider the following restriction of two-prover games:

*What if we demand that Arthur’s challenges to Merlin<sub>1</sub> and Merlin<sub>2</sub> be independent? In other words, what if the distribution  $\mathcal{D}$  is simply the uniform distribution over  $X \times Y$ ?*<sup>3</sup>

---

<sup>2</sup>Because of convexity, we can assume without loss of generality that both Merlins use deterministic strategies.

<sup>3</sup>We could also let  $\mathcal{D}$  be an arbitrary product distribution, but we don’t gain any interesting generality that way: Arthur might as well just send Merlin<sub>1</sub> and Merlin<sub>2</sub> the uniform random bits he would’ve used to generate  $x \in X$  and  $y \in Y$  respectively.

In the PCP literature, two-prover games where  $\mathcal{D}$  is uniform over  $X \times Y$  are called “free” games, and have sometimes been studied as an easier-to-analyze special case of general games [12, 34]. Free games are also tightly connected to dense instances of constraint satisfaction problems. In this paper, we consider approximating the values of free games as an interesting computational problem in its own right, and one that has not received explicit attention. As far as we know, we are the first to study the complexity of this problem directly, and to formulate complexity classes of problems reducible to free games.

In more detail, the restriction to free games gives us an analogue of “public-coin” protocols in the single-prover interactive proof setting. This corresponds to the original definition of the class AM, so we use a version of AM notation. We consider  $\text{AM}(2)$ , or two-prover Arthur-Merlin: the class of all languages that admit two-prover, two-round interactive proof systems, in which Arthur’s challenges to the Merlins are independent, uniformly-random  $\text{poly}(n)$ -bit strings. In other words,  $\text{AM}(2)$  is the class of problems *implicitly* reducible to approximating the value of a free game. Clearly

$$\text{AM} \subseteq \text{AM}(2) \subseteq \text{MIP} = \text{NEXP}. \quad (2)$$

We want to know: *what is the true power of  $\text{AM}(2)$ ?* Is it more powerful than single-prover AM? Is it less powerful than MIP? We will also be interested in the complexity of approximating the values of *explicit* free games.

As we’ll discuss in Section 4, an additional motivation to study  $\text{AM}(2)$  comes from difficult analogous questions about *quantum* multi-prover proof systems, and specifically about the quantum complexity class  $\text{QMA}(2)$ . Our results could shed light on  $\text{QMA}(2)$ , by showing how many questions about it get resolved in a simpler “classical model situation.”

## 2 Our Results

We have two main sets of results: upper bounds, showing that the value of a free game can be approximated in quasipolynomial time and translating that into complexity class containments; and hardness results, giving almost matching lower bounds for this problem under the Exponential Time Hypothesis (ETH). Thus, assuming only the ETH, we show both that free games are exponentially easier than general two-prover games, and *also* they still remain nontrivial, out of reach for polynomial-time algorithms.

### 2.1 Upper Bounds

Let  $\text{FREEGAME}_\varepsilon$  be the problem of approximating the value of a free game to error  $\pm\varepsilon$ :

**Problem 2 ( $\text{FreeGame}_\varepsilon$ )** *Given as input a description of a free game  $G = (X, Y, A, B, V)$ , estimate  $\omega(G)$  to within additive error  $\pm\varepsilon$ . (Here  $n$ , the input size, is  $|X||Y||A||B|$ , and  $\varepsilon$  is an arbitrarily small constant if not specified explicitly.)*

We give a quasipolynomial-time algorithm for  $\text{FREEGAME}_\varepsilon$ :

**Theorem 3**  $\text{FREEGAME}_\varepsilon$  is solvable in deterministic time  $n^{O(\varepsilon^{-2} \log n)}$ .

While this is the first algorithm explicitly for  $\text{FREEGAME}_\varepsilon$ , there is some directly-related algorithmic work. After learning of our results (but before this paper was written), Brandao and

Harrow [15] gave an algorithm for  $\text{FREEGAME}_\epsilon$  with the same running time as ours, but using interestingly different techniques. (Our algorithm is purely combinatorial, whereas theirs uses linear programming relaxation.) Also, Barak et al. [11] gave a quasipolynomial-time approximation algorithm for the related problem of approximating the values of dense CSPs with polynomial-sized alphabets.

In the implicit setting, Theorem 3 implies that  $\text{AM}(2) \subseteq \text{EXP}$ , which improves on the trivial upper bound of  $\text{NEXP}$ . However, by building on the result of Barak et al. [11] mentioned above, we are able to prove a stronger result, which completely characterizes  $\text{AM}(2)$ :

**Theorem 4**  $\text{AM}(2) = \text{AM}$ .

We can even generalize Theorem 4 to handle any polynomial number of Merlins:

**Theorem 5**  $\text{AM}(k) = \text{AM}$  for all  $k = \text{poly}(n)$ .

Thus, in the complexity class setting, it is really the correlation between queries that makes multiple provers more powerful than a single prover.

## 2.2 Hardness Results

Seeing just the above, one might conjecture that the values of free games are approximable in polynomial time. But surprisingly, we give strong evidence that this is *not* the case.

To show the power of free games, we give a nontrivial reduction from 3SAT to  $\text{FREEGAME}$ . Equivalently, we show that *there exists a nontrivial  $\text{AM}(2)$  protocol*: even if Arthur’s challenges are completely independent, two Merlins can be more helpful to him than one Merlin. In particular, given a 3SAT instance  $\varphi$ , let the *size* of  $\varphi$  be the number of variables plus the number of clauses. Then:

**Theorem 6** *For some constant  $\epsilon > 0$ , there exists a reduction running in time  $2^{\tilde{O}(\sqrt{n})}$  that maps 3SAT instances of size  $n$  to  $\text{FREEGAME}_\epsilon$  instances of size  $2^{\tilde{O}(\sqrt{n})}$  (where the  $\tilde{O}$  hides polylogarithmic factors).*

In other words, there is a protocol whereby Arthur can check that a 3SAT instance of size  $n$  is satisfiable, by exchanging only  $\tilde{O}(\sqrt{n})$  bits with the Merlins—i.e., sending  $\tilde{O}(\sqrt{n})$ -bit challenges and receiving  $\tilde{O}(\sqrt{n})$ -bit responses. The protocol has perfect completeness and a  $1$  vs.  $1 - \epsilon$  completeness/soundness gap, for some fixed constant  $\epsilon > 0$ . Since the first step we use is the PCP Theorem, by composing our main protocol with various PCP constructions, we can get reductions with different quantitative tradeoffs between reduction time, completeness, soundness, and alphabet size.

One corollary of Theorem 6 is that, if  $\text{FREEGAME}$  is in  $\text{P}$ , then 3SAT is in  $\text{TIME}(2^{\tilde{O}(\sqrt{n})})$ . Since 3SAT is complete under quasilinear-time reductions for  $\text{NTIME}(n)$ , the same holds for any problem in nondeterministic linear time. As a second corollary, we get a lower bound on the time to approximate  $\text{FREEGAME}$  assuming the ETH. This lower bound almost matches the upper bounds described in Section 2.1. To be more precise, recall the *Exponential Time Hypothesis* (ETH) of Impagliazzo and Paturi [26]:

**Conjecture 7 (Exponential Time Hypothesis [26])** *Any deterministic algorithm for 3SAT requires  $2^{\Omega(n)}$  time. (There is also the Randomized ETH, which says the same for bounded-error randomized algorithms.)*

Then we show the following:

**Corollary 8 (Hardness of Free Games)** *Assuming the (randomized) ETH, any (randomized) algorithm for  $\text{FREEGAME}_\varepsilon$  requires  $n^{\tilde{\Omega}(\varepsilon^{-1} \log n)}$  time, for all  $\varepsilon \geq 1/n$  bounded below some constant.*

Again, by considering various PCP constructions, we get a variety of hardness results for many interesting versions and ranges of parameters for the  $\text{FREEGAME}$  problem.

We can further reduce  $\text{FREEGAME}$  to the problem of approximating dense CSPs, where an arity  $k$  CSP is considered dense if it contains constraints for a constant fraction of all  $k$ -tuples of variables. We thus get the following hardness result for dense CSPs.

**Corollary 9** *Assuming the ETH, the problem of approximating a dense  $k$ -CSP (constraint satisfaction problem) with a polynomial-size alphabet, to constant additive error, requires  $n^{\tilde{\Omega}(\log n)}$  time, for any  $k \geq 2$ .*

Corollary 9 almost matches the upper bound of Barak et al. [11], explaining for the first time why Barak et al. were able to give a quasipolynomial-time algorithm for approximating dense CSPs, but not a polynomial-time one.

As another application of our hardness result for  $\text{FREEGAME}$ , Brandao and Harrow [15] were recently able to use it to prove that approximating the values of certain entangled games requires  $n^{\tilde{\Omega}(\log n)}$  time, assuming the ETH.<sup>4</sup>

### 3 Detailed Overview of Results

We now proceed to more detailed overview of our results and the techniques used to prove them. Here, as in the technical part of the paper, we first describe our hardness results for  $\text{FREEGAME}$  (or equivalently,  $\text{AM}(2)$  protocols for 3SAT), and then our approximation algorithms (or equivalently, limitations of  $\text{AM}(k)$  protocols).

#### 3.1 3Sat Protocol

The idea of our 3SAT protocol is simple. First Arthur transforms the 3SAT instance  $\varphi$  into a PCP, so that it's either satisfiable or far from satisfiable. For this to work, we need a highly-efficient PCP theorem, which produces instances of near-linear size. Fortunately, such PCP theorems are now known. Depending on the desired parameters, we will use either the theorem of Dinur [17] (which produces 3SAT instances of size  $n \text{ polylog } n$  with a small constant completeness/soundness gap), or that of Moshkovitz and Raz [31] (which produces 2-CSP instances of size  $n \cdot 2^{(\log n)^{1-\Omega(1)}}$  with completeness/soundness gap arbitrarily close to 1).

Suppose for now that we use the PCP theorem of Dinur [17]. Then next, Arthur runs a variant of the so-called *clause/variable game*, which we define below.

---

<sup>4</sup>See [15] for the precise definition of the entangled games they consider. Briefly, though, the games involve a large number of provers, of whom two are selected at random to receive challenges (the other provers are ignored).

**Definition 10 (Clause/Variable Game)** Given a 3SAT instance  $\varphi$ , consisting of  $n$  variables  $x_1, \dots, x_n$  and  $m$  clauses  $C_1, \dots, C_m$ , the clause/variable game  $G_\varphi$  is defined as follows. Arthur chooses an index  $i \in [m]$  uniformly at random, then chooses  $j \in [n]$  uniformly at random conditioned on  $x_j$  or  $\neg x_j$  appearing in  $C_i$  as a literal. He sends  $i$  to Merlin<sub>1</sub> and  $j$  to Merlin<sub>2</sub>. Arthur accepts if and only if

- (i) Merlin<sub>1</sub> sends back a satisfying assignment to the variables in  $C_i$ , and
- (ii) Merlin<sub>2</sub> sends back a value for  $x_j$  that agrees with the value sent by Merlin<sub>1</sub>.

Let  $\text{SAT}(\varphi) \in [0, 1]$  be the maximum fraction of clauses of  $\varphi$  that can be simultaneously satisfied. Then clearly the clause/variable game has *perfect completeness*: that is, if  $\text{SAT}(\varphi) = 1$  then  $\omega(G_\varphi) = 1$ . The following well-known proposition shows that the game also has *constant soundness*.

**Proposition 11** If  $\text{SAT}(\varphi) \leq 1 - \varepsilon$ , then  $\omega(G_\varphi) \leq 1 - \varepsilon/3$ .

**Proof.** Assume without loss of generality that Merlin<sub>2</sub> answers according to a particular assignment  $x = (x_1, \dots, x_n)$ . By hypothesis,  $x$  violates the clause  $C_i$  with probability at least  $\varepsilon$  over  $i$ . And if  $x$  violates  $C_i$ , then regardless of what Merlin<sub>1</sub> does, Arthur rejects with probability at least  $1/3$ —since Merlin<sub>1</sub>'s assignment to  $C_i$  either violates  $C_i$ , or else disagrees with  $x$  (which Arthur detects with probability at least  $1/3$  over the variable sent to Merlin<sub>2</sub>). ■

Also, given any two-prover game  $G = (X, Y, A, B, \mathcal{D}, V)$ , let  $G^k$  be the  $k$ -fold parallel repetition of  $G$ : that is, the game where Arthur

- (1) draws  $(x_1, y_1), \dots, (x_k, y_k)$  independently from  $\mathcal{D}$ ,
- (2) sends  $x_1, \dots, x_k$  to Merlin<sub>1</sub> and  $y_1, \dots, y_k$  to Merlin<sub>2</sub>,
- (3) receives responses  $a_1, \dots, a_k \in A$  from Merlin<sub>1</sub> and  $b_1, \dots, b_k \in B$  from Merlin<sub>2</sub>, and then
- (4) accepts with probability equal to  $\prod_{i=1}^k V(x_i, y_i, a_i, b_i)$ .

Then the famous Parallel Repetition Theorem asserts that  $\omega(G^k)$  decreases exponentially with  $k$ :

**Theorem 12 (Parallel Repetition Theorem [33, 25])** If  $\omega(G) \leq 1 - \varepsilon$ , then

$$\omega(G^k) \leq (1 - \varepsilon^3)^{\Omega(k/\log|A||B|)}. \quad (3)$$

Unfortunately, neither the original clause/variable game  $G_\varphi$ , nor its parallel repetition  $G_\varphi^k$ , work in the setting of AM [2]. For both games rely essentially on *correlation* between the clause(s) sent to Merlin<sub>1</sub> and the variable(s) sent to Merlin<sub>2</sub>. To eliminate the need for correlation, we use a new form of repetition that we call *birthday repetition*.

**Definition 13 (Birthday Repetition)** Let  $G = (X, Y, A, B, \mathcal{D}, V)$  be a two-prover game with  $V \in \{0, 1\}$  (not necessarily free). Assume  $\mathcal{D}$  is just the uniform distribution over some subset  $Z \subseteq X \times Y$ . Then given positive integers  $k \leq |X|$  and  $\ell \leq |Y|$ , the birthday repetition  $G^{k \times \ell}$  is the free game defined as follows. Arthur chooses subsets  $S \subseteq X$  and  $T \subseteq Y$  uniformly at random,

subject to  $|S| = k$  and  $|T| = \ell$ . He sends  $S$  to Merlin<sub>1</sub> and asks for an assignment  $a : S \rightarrow A$ , and sends  $T$  to Merlin<sub>2</sub> and asks for an assignment  $b : T \rightarrow B$ . Arthur accepts if and only if  $V(x, y, a(x), b(y)) = 1$  for all  $(x, y) \in S \times T$  that happen to lie in  $Z$ . (So in particular, if  $(S \times T) \cap Z$  is empty, then Arthur always accepts.)

Now consider the birthday repetition  $G_\varphi^{k \times \ell}$  of the clause/variable game  $G_\varphi$ . In this game, Arthur chooses  $k$  random clause indices  $i_1, \dots, i_k$  and sends them to Merlin<sub>1</sub>, and chooses  $\ell$  random variable indices  $j_1, \dots, j_\ell$  and sends them to Merlin<sub>2</sub>. He then sends  $i_1, \dots, i_k$  to Merlin<sub>1</sub> and asks for assignments to  $C_{i_1}, \dots, C_{i_k}$ , and sends  $j_1, \dots, j_\ell$  to Merlin<sub>2</sub> and asks for assignments to  $x_{j_1}, \dots, x_{j_\ell}$ . Finally, Arthur accepts if and only if the assignments to  $C_{i_1}, \dots, C_{i_k}$  satisfy those clauses, *and* are consistent with  $x_{j_1}, \dots, x_{j_\ell}$  on any variables where they happen to intersect.

If  $\varphi$  is satisfiable, then clearly  $\omega(G_\varphi^{k \times \ell}) = 1$ . Our main result says that, if  $\varphi$  is far from satisfiable and  $k, \ell = \Omega(\sqrt{n})$ , then  $\omega(G_\varphi^{k \times \ell}) \leq 1 - \Omega(1)$ . This result is “intuitively plausible,” since if  $k\ell = \Omega(n)$ , then by the Birthday Paradox, there’s a constant probability that some  $x_{j_t}$  will occur as a literal in some  $C_{i_s}$ , giving Arthur a chance to catch the Merlins in an inconsistency if  $\varphi$  is far from satisfiable. But of course, any soundness proof needs to account for the fact that Merlin<sub>1</sub> sees the entire list  $C_{i_1}, \dots, C_{i_k}$ , while Merlin<sub>2</sub> sees the entire list  $x_{j_1}, \dots, x_{j_\ell}$ ! So it’s conceivable that the Merlins could cheat using some clever correlated strategy. We will rule that possibility out, by showing that any cheating strategy for  $G_\varphi^{k \times \ell}$  can be converted (with help from some combinatorial counting arguments) into a cheating strategy for the original clause/variable game  $G_\varphi$ .

One might worry that any proof of a “Birthday Repetition Theorem” would need to be at least as complicated as the proof of the original Parallel Repetition Theorem. Fortunately, though, we can get by with a relatively simple proof, for two reasons. First, we will not prove that birthday repetition works for *every* game  $G$  or for *every*  $k$  and  $\ell$ , for the simple reason that this is false!<sup>5</sup> Instead, our proof will use a special property of the clause/variable game  $G_\varphi$ : namely, the fact that it arises from a uniform constraint graph. Second, we are happy if we can “merely” construct a free game that preserves the soundness of the original game  $G$ : amplifying  $G$ ’s soundness even further would be a bonus, but is not necessary. We leave it to future work to determine the power of birthday repetition more generally.

### 3.2 Approximation Algorithms for Free Games

Our second set of results aims at showing that a square-root savings in communication, as achieved by our AM(2) protocol for 3SAT, is the *best* that any such protocol can provide. More formally, we prove the following set of four interrelated results:

- (1) The FREEGAME <sub>$\varepsilon$</sub>  problem is solvable deterministically in  $(|X| \cdot |A|)^{O(\varepsilon^{-2} \log |Y| |B|)} = n^{O(\varepsilon^{-2} \log n)}$  time. (There is also a randomized algorithm that uses  $|X| \cdot |A|^{O(\varepsilon^{-2} \log |Y| |B|)}$  time.)

---

<sup>5</sup>As a silly counterexample, let  $G$  be the free game with  $X = Y = A = B = [n]$ , where the Merlins lose if and only if  $x = 1$ . Then clearly  $\omega(G) = 1 - 1/n$  and  $\omega(G^{k \times \ell}) = 1 - k/n$ , with no dependence on  $\ell$ . More generally, it is not hard to see that  $\omega(G^{k \times \ell}) \geq \max\{|A|^{-k}, |B|^{-\ell}\}$  for every game  $G$  with  $\omega(G) > 0$ , since this is achieved if one Merlin responds randomly, while the other Merlin *guesses* the first Merlin’s responses and then responds optimally. This implies the following result, for any game  $G$ . Let  $\omega(G^{1 \times 1}) = 1 - \varepsilon$  (note that if  $G$  is free, then  $G^{1 \times 1} = G$ , while otherwise  $G^{1 \times 1}$  is a “promise-free” version of  $G$ ). Then the value  $\omega(G^{k \times \ell})$  can only decrease like  $\omega(G^{1 \times 1})^{\Omega(k\ell)}$  so long as  $k = O(\frac{1}{\varepsilon} \log |B|)$  and  $\ell = O(\frac{1}{\varepsilon} \log |A|)$ .



- (2) Any AM(2) protocol involving  $p(n)$  bits of communication can be simulated in  $2^{O(p(n)^2)}$  poly( $n$ ) time (deterministically, if Arthur’s verification procedure is deterministic, and probabilistically otherwise). So in particular,  $\text{AM}(2) \subseteq \text{EXP}$ , improving the trivial upper bound of NEXP. (As we point out, a closer analysis improves the upper bound to  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ .)
- (3) Assuming the Randomized ETH, any constant-soundness AM(2) protocol for 3SAT must use  $\Omega(\sqrt{n})$  communication. (In more detail, such a protocol must use  $\Omega(\sqrt{\varepsilon n})$  communication if its completeness/soundness gap is 1 vs.  $1 - \varepsilon$ , and  $\Omega(\sqrt{n \log 1/\delta})$  communication if its gap is 1 vs.  $\delta$ . Also, if Arthur’s verification procedure is deterministic, then it suffices to assume the standard ETH.)
- (4)  $\text{AM}(2) = \text{AM}$ . (Of course, this supersedes our  $\text{AM}(2) \subseteq \text{EXP}$  and  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$  results.)

In Section 7.1, we provide a self-contained proof for result (1), and then use (1) to deduce (2) and (3). The idea of our approximation algorithm is to sample a small random subset  $S \subset X$  of the questions to Merlin<sub>1</sub>. We then brute-force search over all possible strategies  $\alpha : S \rightarrow A$  for the questions in  $S$ . For each such strategy  $\alpha$ , we find the optimal response  $b_\alpha : Y \rightarrow B$  of Merlin<sub>2</sub> to that  $\alpha$ , and then the optimal response  $a_\alpha : X \rightarrow A$  of Merlin<sub>1</sub> to  $b_\alpha$  on his full question set  $X$ . A simple probabilistic analysis then shows that, provided we take  $|S| = \Omega(\varepsilon^{-2} \log |Y| |B|)$ , at least one of these “induced” strategy pairs  $(a_\alpha, b_\alpha)$  must achieve value within  $\varepsilon$  of the optimal value  $\omega(G)$ . Similar ideas have been used before in other approximation algorithms: for example, in that of Lipton, Markakis, and Mehta [30] for finding approximate Nash equilibria.

Once we have an  $n^{O(\varepsilon^{-2} \log n)}$ -time approximation algorithm for  $\text{FREEGAME}_\varepsilon$ , the containment  $\text{AM}(2) \subseteq \text{EXP}$  follows almost immediately. We also sketch an improvement to  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ , which is obtained by modifying our approximation algorithm so that it fits into the property-testing framework of Goldreich, Goldwasser, and Ron [21]. As for the optimality of our 3SAT protocol, we simply need to observe that, if we had a protocol that used  $o(\sqrt{n})$  communication, then it would give rise to a free game  $G$  of size  $2^{o(\sqrt{n})}$ , whose value  $\omega(G)$  we could estimate in  $2^{o(n)}$  time by using our quasipolynomial-time approximation algorithm. But that would let us decide 3SAT in  $2^{o(n)}$  time, contradicting the Exponential Time Hypothesis.

For result (4), we wish to go further, and show that any two-Merlin protocol can be simulated using *one* Merlin: that is,  $\text{AM}(2) = \text{AM}$ . Here we appeal to a powerful line of earlier work on *subsampling for dense CSPs*. Specifically, Alon et al. [5] showed in 2002 that, given any  $k$ -ary constraint satisfaction problem  $\varphi$  over  $n$  Boolean variables, one can estimate the maximum number of constraints in  $\varphi$  that can be simultaneously satisfied, to within additive error  $\pm \varepsilon \binom{n}{k}$ , by simply throwing away all the variables except for a random set  $I$  of size  $\text{poly}(1/\varepsilon)$ , and then using brute-force search to find an optimal assignment to  $\varphi_I$ , the restriction of  $\varphi$  to  $I$ .

To build intuition, it is easy to satisfy  $\varphi_I$  *at least as well* as we can satisfy  $\varphi$ , with high probability over  $I$ . To do so, simply start with an optimal global assignment  $x$  for  $\varphi$ ; then restrict  $x$  to the variables in  $I$  and apply a Chernoff bound. The hard part is to show that  $\varphi_I$  cannot be satisfied much *better* than the full instance  $\varphi$  was. Conversely, one needs to show that, given a collection of “local assignments,” involving just  $\text{poly}(1/\varepsilon)$  variables at a time, one can “patch them together” into a global assignment that is almost as good as the local ones.

In later work, Barak et al. [11] proved a more general result, which removed Alon et al.’s assumption that the alphabet is Boolean. Their result lets us approximate the value of any dense  $k$ -CSP  $\varphi$  over the finite alphabet  $\Sigma$  to within additive error  $\pm \varepsilon \binom{n}{k}$ , by solving a random sub-instance on  $\text{poly}(1/\varepsilon) \cdot \log |\Sigma|$  variables.

To see the relevance of this work to free games, we simply need to observe that  $\text{FREEGAME}$  can be directly encoded as a dense CSP. Given a free game  $G = (X, Y, A, B, V)$ , we can create variables  $(a(x))_{x \in X}$  and  $(b(y))_{y \in Y}$  over the alphabets  $A$  and  $B$  respectively, and then for all  $(x, y, a, b) \in X \times Y \times A \times B$ , add a number of constraints setting  $a(x) = a$  and  $b(y) = b$  that is proportional to  $V(x, y, a, b)$ . Once we do this, the result of Barak et al. [11] implies a *subsampling theorem for free games*—saying that the value of any free game  $G$  can be well-approximated by the value of a logarithmic-sized random subgame. And this, in turn, readily implies that  $\text{AM}(2) = \text{AM}$ . For given any  $\text{AM}(2)$  protocol, we can simulate the protocol in  $\text{AM}$  by having Arthur execute the following steps:

- (i) Choose random subsets  $S, T$  of poly( $n$ ) questions to Merlin<sub>1</sub> and Merlin<sub>2</sub> respectively.
- (ii) Ask a *single* Merlin to send him responses to all questions in  $S$  and  $T$ .
- (iii) Check the responses, for all possible question pairs  $(x, y) \in S \times T$ .

The soundness of this approach follows from the subsampling theorem, which says that if Merlins had no winning strategy in the original  $\text{AM}(2)$  protocol, then with high probability, they have no winning strategy even when restricted to the tiny subset of questions  $S \times T$ .

One might ask: if existing results on dense CSPs can be used to show that  $\text{AM}(2) = \text{AM}$ , then why do we “reinvent the wheel,” and provide self-contained proofs for weaker results such as  $\text{AM}(2) \subseteq \text{EXP}$ ? One answer is that the dense CSP results do not give good dependence on the error. For example, those results imply that  $\text{FREEGAME}_\varepsilon$  can be solved in  $n^{O(\varepsilon^{-\Lambda} \log n)}$  time for some large and unspecified constant  $\Lambda$ , but not that it can be solved in  $n^{O(\varepsilon^{-2} \log n)}$  time. And we actually care about the dependence on  $\varepsilon$ , for at least two reasons. First, we wish to make an analogy with a recent  $n^{O(\varepsilon^{-2} \log n)}$  algorithm for a problem in quantum information theory, due to Brandao, Christandl, and Yard [14] (for details see Section 4). And second, we wish to show that, assuming the ETH, the “obvious”  $\text{AM}(2)$  protocol for 3SAT is optimal even in the very low-error and high-error cases. The dense CSP results do not get us close to such a statement, but our algorithm does.

More broadly, appealing to the dense CSP literature feels like overkill if we just want to show (for example) that our 3SAT protocol is optimal, or that the values of free games can be approximated in quasipolynomial time. If we *can* prove those results in an elementary, self-contained way, then it seems like we should—particularly because our proofs might help to make certain striking techniques from the dense CSP world more accessible than they would be otherwise.

Their algorithm also implies that  $\text{AM}(2) \subseteq \text{EXP}$ , and that our 3SAT protocol is essentially optimal assuming the ETH. On the other hand, it seems unlikely that their algorithm can be used to get the containment  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ , let alone  $\text{AM}(2) = \text{AM}$ .

### 3.3 Generalizing to $k$ Merlins

One might wonder whether our limitation theorems for  $\text{AM}(2)$  protocols could be evaded by simply adding more Merlins. So for example, even if  $\text{AM}(2)$  protocols for 3SAT require  $\Omega(\sqrt{n})$  communication (assuming the ETH), could there be an  $\text{AM}(3)$  protocol that used  $O(n^{1/3})$  communication, an  $\text{AM}(10)$  protocol that used  $O(n^{1/10})$  communication, and so forth? In Sections 7.3 and 7.4, we generalize our limitation theorems to the case of  $k$  Merlins, in order to rule out that possibility. In particular, we give the following extensions of our results from Section 3.2:

- (1') There is a deterministic algorithm that, given as input a  $k$ -player free game  $G$  with question sets  $Y_1, \dots, Y_k$  and answer sets  $B_1, \dots, B_k$ , approximates  $\omega(G)$  to within  $\pm\epsilon$  in time

$$\exp\left(\frac{k^2}{\epsilon^2} \sum_{i < j} \log(|Y_i| |B_i|) \cdot \log(|Y_j| |B_j|)\right) = n^{O(\epsilon^{-2} k^2 \log n)}, \quad (4)$$

where  $n = |Y_1| |B_1| \cdots |Y_k| |B_k|$  is the input size. (There is also an alternative algorithm that runs in time  $n^{\epsilon^{-O(1)} \log n}$ , independently of  $k$ .)

- (2')  $\text{AM}(k) \subseteq \text{EXP}$  for all  $k = \text{poly}(n)$ . (Indeed, any constant-soundness  $\text{AM}(k)$  protocol involving  $p(n)$  total bits of communication can be simulated in  $2^{O(p(n)^2)}$   $\text{poly}(n)$  randomized time, or  $2^{O(p(n)^2)}$   $\text{poly}(n)$  deterministic time if Arthur's verification procedure is deterministic.)
- (3') Assuming the Randomized ETH, any constant-soundness  $\text{AM}(k)$  protocol for 3SAT must use  $\Omega(\sqrt{n})$  total bits of communication, regardless of how large  $k$  is. (If, moreover, Arthur's verification procedure is deterministic, then it suffices to assume the ordinary ETH.)
- (4')  $\text{AM}(k) = \text{AM}$  for all  $k = \text{poly}(n)$ .

We first prove (1'), and then derive (2') and (3') as consequences. For (1'), the basic idea is to generalize our approximation algorithm for 2-player free games to  $k$ -player games, by calling the algorithm recursively to “peel off players one at a time.” In other words, we reduce the approximation of a  $k$ -player game to the approximation of a quasipolynomial number of  $(k-1)$ -player games, and continue recursing until we get down to 1 player. When we do this, we need to control the buildup of error across all  $k$  levels of the recursion, and that is why we get a factor of  $k^2$  in the exponent of the running time. Later, by using the subsampling machinery, we will be able to go back and give an alternative algorithm whose running time depends only on  $n$ , not on  $k$ . And that, in turn, will let us show that assuming the ETH, any  $\text{AM}(k)$  protocol for 3SAT must use  $\Omega(\sqrt{n})$  total bits of communication, regardless of  $k$ . (Our first algorithm only implies a lower bound of  $k + \Omega(\sqrt{n}/k) = \Omega(n^{1/4})$  on the total communication, assuming the ETH.) The tradeoff is that the running time of the alternative algorithm depends exponentially on  $\epsilon^{-\Lambda}$  for some large constant  $\Lambda$ , rather than on  $\epsilon^{-2}$ .

For (4'), we need to show that the subsampling theorem of Barak et al. [11] continues to give us what we want, so long as  $k = \text{poly}(n)$ . This boils down to proving a good *subsampling theorem for  $k$ -player free games*. That is, given any  $k$ -player free game  $G = (Y_1, \dots, Y_k, B_1, \dots, B_k, V)$  of total size  $n = |Y_1| |B_1| \cdots |Y_k| |B_k|$ , we need to show that its value  $\omega(G)$  can be approximated to within additive error  $\pm\epsilon$ , by restricting attention to random subsets of questions  $(S_i \subset Y_i)_{i \in [k]}$ , where each  $S_i$  has size  $\epsilon^{-O(1)} \log n$ . A direct adaptation of our argument from the  $k=2$  case turns out not to work here (it breaks down when  $k$  is greater than  $O(\log n)$ ), but we give an alternative encoding of  $k$ -player free games by  $k$ -CSPs that works for all  $k = \text{poly}(n)$ .

## 4 Quantum Motivation

In studying  $\text{AM}(2)$ , our original motivation was to understand the quantum complexity class  $\text{QMA}(2)$  (i.e., two-prover Quantum Merlin-Arthur). So in this section, we provide some background about  $\text{QMA}(2)$ , and explain the tantalizingly close analogy between it and  $\text{AM}(2)$ . Readers who don't care about quantum complexity theory can skip this section.

Recall that “ordinary” QMA is just the quantum analogue of MA:

**Definition 14 (Quantum Merlin-Arthur)** QMA is the class of languages  $L \subseteq \{0,1\}^*$  for which there exists a polynomial-time quantum algorithm  $Q$  such that, for all inputs  $x \in \{0,1\}^n$ :

- If  $x \in L$  then there exists a quantum witness state  $|\phi\rangle$ , on  $\text{poly}(n)$  qubits, such that  $Q(x, |\phi\rangle)$  accepts with probability at least  $2/3$ .
- If  $x \notin L$  then  $Q(x, |\phi\rangle)$  accepts with probability at most  $1/3$ , for all purported witness states  $|\phi\rangle$ .

A lot is known about QMA: for example, it has natural complete promise problems, admits amplification, and is contained in PP (see Aharonov and Naveh [4] for a survey).

Now,  $\text{QMA}(k)$  (introduced by Kobayashi et al. [28]) is just like QMA, but with  $k$  Merlins who are assumed to be unentangled. Note that, if the Merlins were entangled, then the joint state they sent to Arthur could be arbitrary—so from Arthur’s perspective, there might as well be only one Merlin.<sup>6</sup> With  $\text{QMA}(k)$ , the hope is that, ironically, Arthur can exploit his knowledge that the messages are *unentangled* to verify statements that he otherwise could not. More formally:

**Definition 15 ( $k$ -Prover Quantum Merlin-Arthur)**  $\text{QMA}(k)$  is the class of languages  $L \subseteq \{0,1\}^*$  for which there exists a polynomial-time quantum algorithm  $Q$  such that, for all inputs  $x \in \{0,1\}^n$ :

- If  $x \in L$ , then there exist quantum witness states  $|\phi_1\rangle, \dots, |\phi_k\rangle$ , each on  $\text{poly}(n)$  qubits, such that  $Q(x, |\phi_1\rangle \otimes \dots \otimes |\phi_k\rangle)$  accepts with probability at least  $2/3$ .
- If  $x \notin L$  then  $Q(x, |\phi_1\rangle \otimes \dots \otimes |\phi_k\rangle)$  accepts with probability at most  $1/3$  for all purported witness states  $|\phi_1\rangle, \dots, |\phi_k\rangle$ .

Compared to QMA, strikingly little is known about  $\text{QMA}(2)$ . Clearly

$$\text{QMA} \subseteq \text{QMA}(2) \subseteq \text{NEXP}, \quad (5)$$

but we do not know any better containments. We do not even have strong evidence that  $\text{QMA}(2) \neq \text{QMA}$ , or at the other extreme that  $\text{QMA}(2) \neq \text{NEXP}$ . Harrow and Montanaro [23] showed that  $\text{QMA}(2)$  allows exponential amplification of success probabilities, and that  $\text{QMA}(2) = \text{QMA}(k)$  for all  $k \geq 3$ ; even these were surprisingly nontrivial results.

Of course,  $\text{QMA}(2)$  would be of limited interest, if we could never actually *exploit* the promise of unentanglement to do anything new. In 2007, however, Blier and Tapp [13] gave a  $\text{QMA}(2)$  protocol for the NP-complete 3COLORING problem, using two quantum witnesses with only  $\log n$  qubits each. The catch was that Arthur has only a  $1/\text{poly}(n)$  probability of catching the Merlins if they cheat. Even then, however, any one-prover QMA protocol with the same parameters would imply  $\text{NP} \subseteq \text{BQP}$ .

Independently, Aaronson et al. [2] gave a protocol to convince Arthur that a 3SAT instance of size  $n$  is satisfiable, using  $\tilde{O}(\sqrt{n})$  quantum witnesses with  $\log n$  qubits each. Unlike Blier and Tapp’s protocol, Aaronson et al.’s achieved *constant* soundness, and that is why it required more

---

<sup>6</sup>For precisely this reason, in the classical case we trivially have  $\text{MA}(k) = \text{MA}$  for all  $k = \text{poly}(n)$ .

communication ( $\tilde{O}(\sqrt{n})$  rather than  $\log n$ ). Shortly afterward, Aaronson et al.’s protocol was improved by Harrow and Montanaro [23], who showed how to prove 3SAT using two quantum witnesses with  $\tilde{O}(\sqrt{n})$  qubits each; and in a different direction by Chen and Drucker [16], who showed how to measure each of the  $\tilde{O}(\sqrt{n})$  witnesses separately from the others.<sup>7</sup>

Without going into too much detail, all of these  $\tilde{O}(\sqrt{n})$ -qubit protocols for 3SAT ultimately rely on the Birthday Paradox. In particular, they all involve Arthur measuring  $k$  quantum registers with  $\log n$  qubits each—and if we want constant soundness, then (roughly speaking) we need a constant probability that two or more of Arthur’s measurements will reveal information about the same 3SAT variable  $x_j$ . And that is why we need  $k = \Omega(\sqrt{n})$ .

It is tempting to speculate that  $\sqrt{n}$  qubits represents some sort of fundamental barrier for multi-prover QMA protocols: i.e., that assuming we want constant soundness, we can save a quadratic factor in the number of qubits needed to prove 3SAT, but no more than that. Certainly it would be astonishing if 3SAT could be proved (with constant soundness) using two unentangled witnesses with only  $\text{polylog } n$  qubits each. In that case, “scaling up” by an exponential, we would presumably get that  $\text{QMA}(2) = \text{NEXP}$ .

When one thinks about the above questions—or for that matter, almost *any* questions about  $\text{QMA}(2)$ —one is inevitably led to a computational problem that Harrow and Montanaro [23] called the BEST SEPARABLE STATE or BSS problem.

**Problem 16 (BSS<sub>ε</sub>)** *Given as input a Hermitian matrix  $A \in \mathbb{C}^{n^2 \times n^2}$ , with eigenvalues in  $[0, 1]$ , approximate*

$$\lambda_{\text{sep}}(A) := \max_{v, w \in \mathbb{C}^n: \|v\|=\|w\|=1} (v^\dagger \otimes w^\dagger) A (v \otimes w) \quad (6)$$

*to additive error  $\pm \varepsilon$ . (Here  $\varepsilon$  is assumed to be an arbitrarily small constant if not specified otherwise.)*

To build intuition, note that

$$\lambda(A) := \max_{u \in \mathbb{C}^{n^2}: \|u\|=1} u^\dagger A u \quad (7)$$

is just the largest eigenvalue of  $A$ , which is easy to compute. Indeed, the proof of  $\text{QMA} \subseteq \text{PP}$  works by reducing the simulation of a QMA protocol to the computation of  $\lambda(A)$ , for some exponentially-large Hermitian matrix  $A$ .

By contrast, BSS asks us to maximize  $u^\dagger A u$  *only over unit vectors of the form  $u = v \otimes w$* . That is why BSS models the problem of maximizing the verifier’s acceptance probability in a  $\text{QMA}(2)$  protocol, where the maximum is taken over all separable witnesses, of the form  $|\phi_1\rangle \otimes |\phi_2\rangle$ . From this standpoint, the reason why  $\text{QMA}(2)$  is so much harder to understand than QMA—but also why  $\text{QMA}(2)$  is potentially more powerful—is that (as one can check) BSS is a non-convex optimization problem, which lacks the clean linear-algebraic structure of computing  $\lambda(A)$ .

Indeed, from the protocol of Blier and Tapp [13] mentioned earlier, it follows immediately that we can reduce 3COLORING to the problem of approximating  $\lambda_{\text{sep}}(A)$  up to additive error  $\pm 1/\text{poly}(n)$ . Furthermore, since the quantum witnesses in the Blier-Tapp protocol have only  $\log n$  qubits, the resulting matrix  $A$  will have size  $2^{O(\log n)} = \text{poly}(n)$ . Thus:

---

<sup>7</sup>It is still not known whether one can combine the Harrow-Montanaro and Chen-Drucker improvements, to get a 3SAT protocol using two witnesses of  $\tilde{O}(\sqrt{n})$  qubits each that are measured separately from each other.

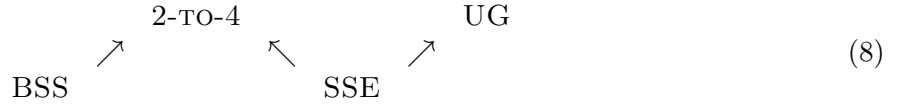
**Theorem 17 (Blier and Tapp [13])**  $\text{BSS}_{1/\text{poly}(n)}$  is NP-hard.

One wants to know: is  $\text{BSS}_\varepsilon$  still a hard problem even for *constant*  $\varepsilon$ ? Because it has constant soundness, the protocol of Harrow and Montanaro [23] (building on Aaronson et al. [2]) lets us reduce 3SAT to the problem of approximating  $\lambda_{\text{sep}}(A)$  up to constant additive error. Now, since the quantum witnesses in the Harrow-Montanaro protocol have  $\tilde{O}(\sqrt{n})$  qubits, the resulting matrix  $A$  has size  $2^{\tilde{O}(\sqrt{n})}$ , so we do not get a polynomial-time reduction. We do, however, get something:

**Theorem 18** *If BSS is solvable in  $t(n)$  time, then 3SAT is solvable in  $t(2^{\tilde{O}(\sqrt{n})})$  time. So in particular, assuming the Exponential Time Hypothesis, BSS requires  $n^{\Omega(\log n)}$  deterministic time. (Likewise, assuming the Randomized ETH, BSS requires  $n^{\Omega(\log n)}$  randomized time.)*

Could we go further than Theorems 17 and 18, and prove that  $\text{BSS}_\varepsilon$  is NP-hard even for constant  $\varepsilon$ ? Notice that if we could, then “scaling up by an exponential,” we could presumably also show  $\text{QMA}(2) = \text{NEXP}$ ! If, on the other hand, we believe (as seems plausible) that  $\text{QMA}(2) \subseteq \text{EXP}$ , then we seem forced to believe that BSS is solvable in  $n^{\text{polylog } n}$  time, even if we have no idea what the algorithm is.<sup>8</sup>

Raising the stakes even further, Barak et al. [10] showed that BSS is intimately related to other problems of great current interest in complexity theory: namely, the UNIQUE GAMES, SMALL SET EXPANSION, and 2-TO-4 NORM problems.<sup>9</sup> The lattice of known reductions among these problems is as follows:



Now, assuming the ETH, Theorem 18 gives us  $n^{\Omega(\log n)}$  hardness for BSS—and as a direct consequence, for 2-TO-4 NORM as well. That might not sound like much, but it’s a lot more than we currently know for either UNIQUE GAMES or SMALL SET EXPANSION! So the speculation arises that, if we fully understood BSS, we might be able to apply some of the insights to UG or SSE.

To lay our cards on the table, here is our conjecture about BSS:

**Conjecture 19**  $\text{BSS}_\varepsilon$  is solvable in deterministic  $n^{O(\varepsilon^{-2} \log n)}$  time.

If true, Conjecture 19 readily implies that  $\text{QMA}(2) \subseteq \text{EXP}$ . Since a  $t(n)$ -time algorithm for BSS can be combined with a  $q(n)$ -qubit  $\text{QMA}(2)$  protocol for 3SAT to get a  $t(2^{O(q(n))})$ -time algorithm for 3SAT, Conjecture 19 also implies that, assuming the ETH, any  $\text{QMA}(2)$  protocol for 3SAT must use  $\Omega(\sqrt{n})$  qubits.

<sup>8</sup>Strictly speaking, neither of these implications is a theorem. For example, even if  $\text{BSS}_\varepsilon$  turned out to be NP-hard for constant  $\varepsilon$ , it’s possible that one could exploit the special structure of the matrices arising from polynomial-size quantum circuits to show that  $\text{QMA}(2) \subseteq \text{EXP}$ . In practice, however, a “reasonable” proof that  $\text{BSS}_\varepsilon$  is NP-hard would probably also imply  $\text{QMA}(2) = \text{NEXP}$ , and a “reasonable” proof of  $\text{QMA}(2) \subseteq \text{EXP}$  would probably proceed by solving  $\text{BSS}_\varepsilon$  in quasipolynomial time.

<sup>9</sup>We refer the reader to [10] for the definitions of these problems, but very briefly: UNIQUE GAMES (UG) is the problem of deciding whether  $\omega(G)$  is close to 1 or close to 0, given as input a description of a two-prover game  $G = (X, Y, A, B, \mathcal{D}, V)$  with the special properties that  $|A| = |B|$ , and that for every  $(x, y) \in X \times Y$  there exists a permutation  $\pi_{x,y}$  such that  $V(x, y, a, b) = 1$  if and only if  $b = \pi_{x,y}(a)$ . SMALL SET EXPANSION (SSE) is the problem of deciding whether a given graph  $G$  is close to or far from an expander graph, if we consider  $G$ ’s expansion on “small” subsets of vertices only. 2-TO-4 NORM (2-TO-4) is the problem, given as input an  $n \times n$  matrix  $A$ , of approximating the maximum of  $\|Av\|_4$  over all vectors  $v$  such that  $\|v\|_2 = 1$ .

There has been some progress toward a proof of Conjecture 19. In particular, Brandao, Christandl, and Yard [14] gave an algorithm that solves  $\text{BSS}_\varepsilon$  in  $n^{O(\varepsilon^{-2} \log n)}$  time if  $\|A\|_2 = O(1)$ , or alternatively, if  $A$  represents a quantum measurement that can be implemented using LOCC (Local Operations and Classical Communication). This implied, among other things, that  $\text{QMA}_{\text{LOCC}}(k) = \text{QMA}$  for  $k = O(1)$ , where  $\text{QMA}_{\text{LOCC}}(k)$  is the subclass of  $\text{QMA}(k)$  in which Arthur is restricted to LOCC measurements. Brandao et al.’s algorithm uses a technique that quantum information researchers know as *symmetric extension*, and that theoretical computer scientists know as the *Lasserre hierarchy*. It is not known whether similar techniques could work for arbitrary  $\text{QMA}(k)$  protocols.

More recently, Brandao and Harrow [15] showed that, assuming the ETH, any so-called  $\text{BellQMA}(k)$  protocol for 3SAT—that is, any  $\text{QMA}(k)$  protocol where each of the  $k$  witnesses are measured separately—must use  $n^{1/2-o(1)}$  qubits. This lower bound is known to be essentially tight, due to the protocol of Chen and Drucker [16]. The requirement that each witness be measured separately (with the measurement outcomes then combined with classical postprocessing) is even more stringent than the requirement of LOCC. Despite this, the result of Brandao and Harrow [15] did not follow from the earlier result of Brandao, Christandl, and Yard [14] that  $\text{QMA}_{\text{LOCC}}(k) = \text{QMA}$ , because the latter works only for constant  $k$ .

## 4.1 Connection to Our Results

But what does any of the above have to do with  $\text{AM}(2)$ ? One way to view this paper’s contribution is as follows: *we prove that a “classical analogue” of Conjecture 19 holds*. In more detail, we can think of  $\text{AM}(2)$  as closely analogous in many ways to  $\text{QMA}(2)$ . For both classes, the only obvious lower bound comes from restricting to a single Merlin, while the only obvious upper bound is  $\text{NEXP}$ . For both classes, the difficulty with proving an  $\text{EXP}$  upper bound is the requirement that the Merlins can’t communicate, which gives rise to a non-convex optimization problem. For both classes, there exists a protocol for 3SAT that uses  $\log n$  communication, but that has only a  $1/\text{poly}(n)$  probability of catching cheating Merlins. For both classes, we can improve the 3SAT protocol to have constant soundness, by using a strong PCP theorem together with the Birthday Paradox—but if we do so, then the communication cost increases from  $\log n$  to  $\tilde{O}(\sqrt{n})$ .

Because the analogy runs so deep, it seems of interest to  $\text{QMA}(2)$  researchers to know that:

- (1)  $\text{FREEGAME}_\varepsilon$  is solvable in  $n^{O(\varepsilon^{-2} \log n)}$  time, as we conjecture that  $\text{BSS}_\varepsilon$  is.
- (2)  $\text{AM}(2)$  is contained in  $\text{EXP}$ , as we conjecture that  $\text{QMA}(2)$  is.
- (3) The  $\tilde{O}(\sqrt{n})$ -communication  $\text{AM}(2)$  protocol for 3SAT is essentially optimal assuming the ETH, as we conjecture that the corresponding  $\text{QMA}(2)$  protocol is.

Of course, we also show in this paper that  $\text{AM}(2) = \text{AM}$ . So pushing the analogy between  $\text{AM}(2)$  and  $\text{QMA}(2)$  all the way to the end would lead to the conjecture that  $\text{QMA}(2) = \text{QMA}$ . We remain agnostic about whether the analogy extends *that* far!

## 5 Preliminaries

Some notation: we use  $\mathbb{E}$  for expectation,  $[n]$  for  $\{1, \dots, n\}$ , and  $\binom{[n]}{k}$  for the set of subsets of  $[n]$  of size  $k$ . In addition to the notation  $\tilde{O}(f(n))$  for  $O(f(n) \text{polylog } f(n))$ , we also use  $\tilde{\Omega}(f(n))$  for

$\Omega(f(n) / \text{polylog } f(n))$ . All logs are base 2 unless specified otherwise.

Sections 1 and 2 have already defined many of the concepts we will need, including two-prover games, free games, the clause/variable game, the birthday repetition, and the FREEGAME problem. For completeness, though, we now give the general definition of  $k$ -player free games.

**Definition 20 ( $k$ -Player Free Games)** *A  $k$ -player free game  $G$  consists of:*

- (1) *finite question sets  $Y_1, \dots, Y_k$  and answer sets  $B_1, \dots, B_k$ , and*
- (2) *a verification function  $V : Y_1 \times \dots \times Y_k \times B_1 \times \dots \times B_k \rightarrow [0, 1]$ .*

*The value of the game, denoted  $\omega(G)$ , is the maximum, over all tuples of response functions  $(b_i : Y_i \rightarrow B_i)_{i \in [k]}$ , of*

$$\mathbb{E}_{y_1 \in Y_1, \dots, y_k \in Y_k} [V(y_1, \dots, y_k, b_1(y_1), \dots, b_k(y_k))]. \quad (9)$$

Directly related to  $k$ -player free games is the complexity class  $\text{AM}(k)$ , which we now formally define.<sup>10</sup>

**Definition 21 ( $k$ -Prover Arthur-Merlin)** *Let  $k$  be a positive integer. Then  $\text{AM}(k)$  is the class of languages  $L \subseteq \{0, 1\}^*$  for which there exists a probabilistic polynomial-time verifier  $V$  such that for all  $n$  and all inputs  $x \in \{0, 1\}^n$ :*

- **(Completeness)** *If  $x \in L$ , then there exist functions  $b_1, \dots, b_k : \{0, 1\}^{\text{poly}(n)} \rightarrow \{0, 1\}^{\text{poly}(n)}$ , depending on  $x$ , such that*

$$\Pr_{y_1, \dots, y_k \in_R \{0, 1\}^{\text{poly}(n)}} [V(x, y_1, \dots, y_k, b_1(y_1), \dots, b_k(y_k)) \text{ accepts}] \geq \frac{2}{3}. \quad (10)$$

- **(Soundness)** *If  $x \notin L$ , then for all such functions  $b_1, \dots, b_k$ ,*

$$\Pr_{y_1, \dots, y_k \in_R \{0, 1\}^{\text{poly}(n)}} [V(x, y_1, \dots, y_k, b_1(y_1), \dots, b_k(y_k)) \text{ accepts}] \leq \frac{1}{3}. \quad (11)$$

Clearly  $\text{AM}(1) = \text{AM}$  and  $\text{AM}(k) \subseteq \text{AM}(k+1)$  for all  $k$ . We also have  $\text{AM}(k) \subseteq \text{MIP}(k)$ , thereby giving the crude upper bound  $\text{AM}(k) \subseteq \text{NEXP}$  (later we will do much better).

We can easily generalize the definition of  $\text{AM}(k)$  to  $\text{AM}(k(n))$ , for any growth rate  $k(n) = O(\text{poly}(n))$ . Also, let  $\text{AM}_{p(n)}(k)$  be the variant of  $\text{AM}(k)$  where all messages (both the  $y_i$ 's and the  $b_i$ 's) are constrained to be  $p(n)$  bits long.

Given any probabilistic complexity class  $\mathcal{C}$ , one of the first questions we can ask is whether  $\mathcal{C}$  admits amplification of success probabilities—or equivalently, whether  $\mathcal{C}$  is robust under changing its error parameters (such as  $1/3$  and  $2/3$ ). At least for  $\text{AM}(2)$ , we are fortunate that a positive answer follows from known results. In particular, building on the Parallel Repetition Theorem (Theorem 12), Rao [32] proved a useful concentration bound for the parallel repetitions of two-prover games:

---

<sup>10</sup> $\text{AM}(k)$  should not be confused with  $\text{AM}[k]$ , which means  $\text{AM}$  with a single Merlin but  $k$  rounds of communication. A classic result of Babai and Moran [9] says that  $\text{AM}[k] = \text{AM}[2] = \text{AM}$  for all constants  $k \geq 2$ . When  $k = \text{poly}(n)$ , by contrast, such a collapse is not believed to happen, since  $\text{AM}[\text{poly}] = \text{IP} = \text{PSPACE}$ .



**Theorem 22 (Rao’s Concentration Theorem [32])** *For all  $\delta > 0$  and all two-prover games  $G = (X, Y, A, B, \mathcal{D}, V)$ , if Merlin<sub>1</sub> and Merlin<sub>2</sub> play the parallel repeated version  $G^N$ , then they can win more than a  $\omega(G) + \delta/4$  fraction of the games with probability at most*

$$2 \left( 1 - \frac{\delta/2}{\omega(G) + 3\delta/4} \right)^{\Omega\left(\frac{\delta^2 N}{\log|A||B| - \log(\omega(G) + \delta/4)}\right)} \quad (12)$$

Theorem 22 implies that “amplification works” for AM(2) protocols:

**Proposition 23** *In the definition of AM(2), replacing the constants  $(1/3, 2/3)$  by  $(a, b)$  for any constants  $0 < a < b < 1$ , or indeed by  $(2^{-p(n)}, 1 - 2^{-p(n)})$  or  $(1/2 - 1/p(n), 1/2 + 1/p(n))$  for any polynomial  $p$ , gives rise to the same complexity class.*

**Proof.** Suppose, for example, that we want to amplify  $(1/3, 2/3)$  to  $(2^{-p(n)}, 1 - 2^{-p(n)})$ ; the other cases are analogous. Given a language  $L \in \text{AM}(2)$  and an input  $x \in \{0, 1\}^n$ , the AM(2) protocol for checking whether  $x \in L$  can be represented as a free game  $G = (X, Y, A, B, V)$ , where  $X = Y = A = B = \{0, 1\}^{q(n)}$  for some polynomial  $q$ . We have  $\omega(G) \geq 2/3$  if  $x \in L$ , and  $\omega(G) \leq 1/3$  if  $x \notin L$ . Now let  $G_{1/2}^N$  be the game where the Merlins play  $N$  parallel instances of the original game  $G$ , and they “win” if and only if they win on at least  $N/2$  instances. If  $\omega(G) \geq 2/3$ , then clearly  $\omega(G_{1/2}^N) \geq 1 - 2^{-\Omega(N)}$ —since if the Merlins just play their optimal strategy for  $G$  on each of the  $N$  instances separately, then the number that they win will be concentrated around  $\omega(G)N \geq 2N/3$  by a standard Chernoff bound. On the other hand, if  $\omega(G) \leq 1/3$ , then Theorem 22 implies that  $\omega(G_{1/2}^N) \leq 2^{-\Omega(N/q(n))}$ . So, by simply choosing  $N \gg p(n)q(n)$  to be a suitably large polynomial, we can ensure that  $\omega(G_{1/2}^N) \geq 1 - 2^{-p(n)}$  if  $x \in L$  while  $\omega(G_{1/2}^N) \leq 2^{-p(n)}$  if  $x \notin L$ . ■

Note that Proposition 23 can blow up the communication cost by a polynomial factor, because of the dependence of  $N$  on  $q(n)$  (which derives from the  $1/\log|A||B|$  factor in the exponent from Theorem 22). For this reason, Proposition 23 doesn’t directly imply any useful amplification for our  $\tilde{O}(\sqrt{n})$ -communication protocol for 3SAT. See Section 6.3 for further discussion of this issue, and for our best current results for the low-error case.

Rao (personal communication) believes that it would be straightforward to generalize Theorem 22 to games with  $k \geq 3$  players, as long as the games are free.<sup>11</sup> If so, then we would also obtain an amplification theorem for AM( $k$ ), for all  $k = \text{poly}(n)$ . However, this generalization has not yet been worked out explicitly.

One last remark: a classic result about “ordinary” AM (see [20]) states that any AM protocol can be made to have *perfect completeness*. In other words, the condition  $x \in L \Rightarrow \Pr[V \text{ accepts}] \geq 2/3$  can be strengthened to  $x \in L \Rightarrow \Pr[V \text{ accepts}] = 1$  without loss of generality. Another classic result [22] states that any AM protocol can be made *public-coin*, meaning that any random bits generated by Arthur are immediately shared with Merlin. In terms of games, the public-coin property would imply in particular that Arthur’s verification function was deterministic: that is,  $V(x, y, a, b) \in \{0, 1\}$  for all  $x, y, a, b$ .

Thus, one might wonder whether any AM( $k$ ) protocol can be made perfect-completeness and public-coin as well. Ultimately, affirmative answers to these questions will follow from our result

<sup>11</sup>By contrast, for *general* games with  $k \geq 3$  players, even proving a “standard” parallel repetition theorem is a notorious open problem.

that  $\text{AM}(k) = \text{AM}$ , which works regardless of whether the original  $\text{AM}(k)$  protocol had perfect completeness or was public-coin. But it would be interesting to find *direct* proofs of these properties for  $\text{AM}(k)$ . (It would also be interesting to find a direct proof that  $\text{AM}(k) = \text{AM}(2)$  for all  $k > 2$ , rather than deducing this as a consequence of  $\text{AM}(k) = \text{AM}$ .)

## 6 Analysis of the Birthday Game

Our goal in this section is to prove Theorem 6: informally, that  $\text{AM}(2)$  protocols for 3SAT can achieve nearly a quadratic savings in communication over the “naïve” bound of  $n$  bits. The section is organized as follows. First, in Section 6.1, we give a “basic” protocol with a  $1$  vs.  $1 - \epsilon$  completeness/soundness gap (for some fixed  $\epsilon > 0$ ) and  $\tilde{O}(\sqrt{n})$  communication cost. The protocol is based on the birthday repetition already discussed in Section 3.1; for concreteness, we initially implement the idea using Dinur’s PCP Theorem and the clause/variable game. Next, in Section 6.2, we study the high-error case, showing how a more refined analysis leads to a protocol with a  $1$  vs.  $1 - \epsilon$  completeness/soundness gap and  $O(\sqrt{\epsilon n} \text{polylog } n)$  communication cost. Then in Section 6.3, we switch to the low-error case, using the PCP Theorem of Moshkovitz and Raz [31] to obtain an  $\text{AM}(2)$  protocol for 3SAT with a  $1$  vs.  $\delta$  completeness/soundness gap and  $n^{1/2+o(1)} \text{poly}(1/\delta)$  communication cost. Finally, in Section 6.4, we give the implication  $\text{NTIME}[n] \subseteq \text{AM}_{n^{1/2+o(1)}}(2)$  and show that this implication is nonrelativizing.

### 6.1 The Basic Result

The first step is to state a variant of the PCP Theorem that is strong enough for our purposes.

**Theorem 24 (PCP Theorem, Dinur’s Version [17])** *Given a 3SAT instance  $\varphi$  of size  $n$ , it is possible in  $\text{poly}(n)$  time to produce a new 3SAT instance  $\phi$ , of size  $n \text{polylog } n$ , such that:*

- **(Completeness)** *If  $\text{SAT}(\varphi) = 1$  then  $\text{SAT}(\phi) = 1$ .*
- **(Soundness)** *If  $\text{SAT}(\varphi) < 1$  then  $\text{SAT}(\phi) < 1 - \epsilon$ , for some constant  $0 < \epsilon < 1/8$ .*
- **(Balance)** *Every clause of  $\phi$  involves exactly 3 variables, and every variable of  $\phi$  appears in exactly  $d$  clauses, for some constant  $d$ .<sup>12</sup>*

The reason why, for now, we use Dinur’s version of the PCP Theorem is that it produces instances of size  $n \text{polylog } n$ . Later, in Section 6.3, we will switch over to the PCP Theorem of Moshkovitz and Raz [31], which produces instances of the slightly larger size  $n \cdot 2^{(\log n)^{1-\Delta}} = n^{1+o(1)}$  (for some constant  $\Delta > 0$ ) but achieves sub-constant error. Were we willing to accept a protocol with  $\sqrt{n} 2^{(\log n)^{1-\Delta}}$  communication, we could have used the Moshkovitz-Raz version from the start, but we will try to keep the communication cost down to  $\sqrt{n} \text{polylog } n$  for as long as we can.

Let the 3SAT instance  $\phi$  produced by Theorem 24 have  $N$  variables  $x_1, \dots, x_N$  and  $M$  clauses  $C_1, \dots, C_M$ . Also, let  $G_\phi$  be the clause/variable game for  $\phi$ , as defined in Section 3.1. Then combining Theorem 24 with Proposition 11 yields the following corollary.

**Corollary 25** *If  $\phi$  is unsatisfiable, then  $\omega(G_\phi) < 1 - \epsilon/3$ .*

---

<sup>12</sup>It is known that we can assume this balance condition without loss of generality.

Next, given positive integers  $k$  and  $\ell$ , let  $G_\phi^{k \times \ell}$  be the birthday repetition of  $\phi$ , also defined in Section 3.1. Then to prove Theorem 6, it suffices to show that  $\omega(G_\phi^{k \times \ell})$  is bounded away from 1, assuming that  $\phi$  is unsatisfiable and that  $k\ell = \Omega(N)$ .

Our strategy for upper-bounding  $\omega(G_\phi^{k \times \ell})$  will be to relate it to  $\omega(G_\phi)$ , which we already know is bounded away from 1. More concretely:

**Theorem 26** *For all  $k \in [M]$  and  $\ell \in [N]$ ,*

$$\omega(G_\phi) \geq \omega(G_\phi^{k \times \ell}) - O\left(\sqrt{\frac{N}{k\ell}}\right). \quad (13)$$

So in particular, by choosing  $k = \ell = c\sqrt{N}$ , where  $c$  is some sufficiently large constant, we can ensure (say)  $\omega(G_\phi^{k \times \ell}) \leq \omega(G_\phi) + 0.01$ .

Let  $\mathcal{U}$  be the uniform distribution over all input pairs

$$(I, J) \in \binom{[M]}{k} \times \binom{[N]}{\ell}, \quad (14)$$

and let  $V_{BD}$  be Arthur's verification function in  $G_\phi^{k \times \ell}$ . To prove Theorem 26, we consider an arbitrary cheating strategy for Merlin<sub>1</sub> and Merlin<sub>2</sub> in the birthday game:

$$a : \binom{[M]}{k} \rightarrow \left(\{0, 1\}^3\right)^k, \quad b : \binom{[N]}{\ell} \rightarrow \{0, 1\}^\ell. \quad (15)$$

Let  $p$  be the success probability of that cheating strategy: that is,

$$p = \mathbb{E}_{(I, J) \sim \mathcal{U}} [V_{BD}(I, J, a(I), b(J))]. \quad (16)$$

Using  $a$  and  $b$ , our task is to construct a cheating strategy for the original clause/variable game  $G_\phi$ , which succeeds with probability at least  $p - O(\sqrt{N/k\ell})$ . That strategy will be the “natural” one: namely, given as input a clause index  $i \in [M]$ , Merlin<sub>1</sub> first chooses a subset  $\{i_1, \dots, i_{k-1}\}$  uniformly at random from  $\binom{[M] - \{i\}}{k-1}$ , and sets  $I := \{i, i_1, \dots, i_{k-1}\}$ . (Crucially,  $I$  is a set, so if its elements were listed in some canonical way—for example, in order— $i$  would generally be somewhere in the middle, and would not be particularly conspicuous!) Merlin<sub>1</sub> then computes  $a(I) \in \left(\{0, 1\}^3\right)^k$ , and sends Arthur the restriction of  $a(I)$  to the index  $i$ . Likewise, given as input a variable index  $j \in [N]$ , Merlin<sub>2</sub> first chooses a subset  $\{j_1, \dots, j_{\ell-1}\}$  uniformly at random from  $\binom{[N] - \{j\}}{\ell-1}$ , and sets  $J := \{j, j_1, \dots, j_{\ell-1}\}$ . He then computes  $b(J) \in \{0, 1\}^\ell$ , and sends Arthur the restriction of  $b(J)$  to the index  $j$ . Of course, the resulting strategy is randomized, but we can convert it to an equally-good deterministic strategy using convexity.

Let  $\mathcal{D}$  be the probability distribution over  $(I, J)$  pairs induced by the cheating strategy above, if we average over all valid inputs  $(i, j)$  to the original clause/variable game. Then let  $q$  be the Merlins' success probability in the birthday game, if they use their same cheating strategy  $(a, b)$ , but for  $(I, J)$  pairs drawn from  $\mathcal{D}$ , rather than from the uniform distribution  $\mathcal{U}$ :

$$q = \mathbb{E}_{(I, J) \sim \mathcal{D}} [V_{BD}(I, J, a(I), b(J))]. \quad (17)$$

Clearly the Merlins' success probability in the clause/variable game is at least  $q$ , since any time they win  $G_\phi^{k \times \ell}$ , they also win its restriction to  $G_\phi$ . Therefore, to prove Theorem 26, it suffices to prove that  $q \geq p - O(\sqrt{N/k\ell})$ . And to do *that*, it in turn suffices to show that  $\mathcal{D}$  is close in variation distance to the uniform distribution  $\mathcal{U}$ , since

$$\left| \mathbb{E}_{\mathcal{D}}[Z] - \mathbb{E}_{\mathcal{U}}[Z] \right| \leq \|\mathcal{D} - \mathcal{U}\| \quad (18)$$

for any  $[0, 1]$  random variable  $Z$ . We upper-bound  $\|\mathcal{D} - \mathcal{U}\|$  in the following lemma.

**Lemma 27**  $\|\mathcal{D} - \mathcal{U}\| = O\left(\sqrt{\frac{N}{k\ell}}\right)$ .

**Proof.** Let  $A = (a_{ij}) \in \{0, 1\}^{M \times N}$  be the incidence matrix of the 3SAT instance  $\phi$ . That is, set  $a_{ij} := 1$  if the clause  $C_i$  involves the variable  $x_j$ , and  $a_{ij} := 0$  otherwise. Note that, by the balance condition, we must have  $\sum_{ij} a_{ij} = 3M = dN$  (where  $d$  is the number of clauses that each variable appears in), and

$$\sum_{j \in [N]} a_{ij} = 3 \quad \forall i, \quad \sum_{i \in [M]} a_{ij} = d \quad \forall j. \quad (19)$$

Given any  $I \subseteq [M]$  and  $J \subseteq [N]$ , define

$$S_{IJ} := \sum_{i \in I, j \in J} a_{ij}, \quad (20)$$

and observe that

$$\mathbb{E}_{|I|=k, |J|=\ell} [S_{IJ}] = \frac{3k\ell}{N}. \quad (21)$$

In the clause/variable game  $G_\phi$ , Arthur chooses his input  $(i, j) \in [M] \times [N]$  uniformly at random subject to  $a_{ij} = 1$ . Now consider an  $(I, J)$  drawn from  $\mathcal{D}$ . By symmetry,  $(I, J)$  is equally likely to have been formed starting from any input  $(i, j) \in I \times J$  such that  $a_{ij} = 1$ . This means that  $\Pr_{\mathcal{D}}[(I, J)]$  must simply be proportional to  $S_{IJ}$ , and normalization gives us the rest:

$$\Pr_{\mathcal{D}}[(I, J)] = \Pr_{\mathcal{U}}[(I, J)] \cdot \frac{S_{IJ}}{3k\ell/N}. \quad (22)$$

Thus,

$$\|\mathcal{D} - \mathcal{U}\| = \frac{1}{2} \sum_{|I|=k, |J|=\ell} \left| \Pr_{\mathcal{D}}[(I, J)] - \Pr_{\mathcal{U}}[(I, J)] \right| \quad (23)$$

$$= \frac{1}{2} \mathbb{E}_{|I|=k, |J|=\ell} \left[ \left| \frac{S_{IJ}}{3k\ell/N} - 1 \right| \right] \quad (24)$$

$$\leq \frac{1}{2} \sqrt{\mathbb{E}_{|I|=k, |J|=\ell} \left[ \left( \frac{S_{IJ}}{3k\ell/N} - 1 \right)^2 \right]} \quad (25)$$

$$= \frac{1}{2} \sqrt{\frac{\mathbb{E}_{|I|=k, |J|=\ell} [S_{IJ}^2]}{(3k\ell/N)^2} - 1} \quad (26)$$

where line (25) used Cauchy-Schwarz. Now,

$$\mathbb{E}_{|I|=k, |J|=\ell} [S_{IJ}^2] = \mathbb{E}_{|I|=k, |J|=\ell} \left[ \sum_{i, i' \in I, j, j' \in J} a_{ij} a_{i'j'} \right] \quad (27)$$

$$= \sum_{i, i' \in [M], j, j' \in [N]} a_{ij} a_{i'j'} \Pr_{|I|=k, |J|=\ell} [i, i' \in I, j, j' \in J]. \quad (28)$$

Here it is convenient to divide the sum into four cases: the case  $i = i'$  and  $j = j'$ , the case  $i = i'$  but  $j \neq j'$ , the case  $j = j'$  but  $i \neq i'$ , and the case  $i \neq i'$  and  $j \neq j'$ . These cases give us respectively:

$$\sum_{i \in [M], j \in [N]} a_{ij} \Pr_{|I|=k, |J|=\ell} [i, i' \in I, j, j' \in J] \leq 3M \frac{k\ell}{MN}, \quad (29)$$

$$\sum_{i \in [M], j \neq j' \in [N]} a_{ij} a_{ij'} \Pr_{|I|=k, |J|=\ell} [i, i' \in I, j, j' \in J] \leq 6M \frac{k\ell(\ell-1)}{MN(N-1)}, \quad (30)$$

$$\sum_{i \neq i' \in [M], j \in [N]} a_{ij} a_{i'j} \Pr_{|I|=k, |J|=\ell} [i, i' \in I, j, j' \in J] \leq d(d-1)N \frac{k(k-1)\ell}{M(M-1)N}, \quad (31)$$

$$\sum_{i \neq i' \in [M], j \neq j' \in [N]} a_{ij} a_{i'j'} \Pr_{|I|=k, |J|=\ell} [i, i' \in I, j, j' \in J] \leq (3M)^2 \frac{k(k-1)\ell(\ell-1)}{M(M-1)N(N-1)}. \quad (32)$$

Hence

$$\mathbb{E}_{|I|=k, |J|=\ell} [S_{IJ}^2] \quad (33)$$

$$\leq 3M \frac{k\ell}{MN} + 6M \frac{k\ell(\ell-1)}{MN(N-1)} + d(d-1)N \frac{k(k-1)\ell}{M(M-1)N} + (3M)^2 \frac{k(k-1)\ell(\ell-1)}{M(M-1)N(N-1)} \quad (34)$$

$$= \left( \frac{3k\ell}{N} \right)^2 \left( O\left( \frac{N}{k\ell} \right) + O\left( \frac{1}{k} \right) + O\left( \frac{1}{\ell} \right) + 1 \right) \quad (35)$$

$$= \left( \frac{3k\ell}{N} \right)^2 \left( 1 + O\left( \frac{N}{k\ell} \right) \right), \quad (36)$$

where we treated  $d$  as a constant. Therefore

$$\|\mathcal{D} - \mathcal{U}\| \leq \frac{1}{2} \sqrt{\frac{\mathbb{E}_{|I|=k, |J|=\ell} [S_{IJ}^2]}{(3k\ell/N)^2} - 1} = O\left( \sqrt{\frac{N}{k\ell}} \right). \quad (37)$$

■

This completes the proof of Theorem 26—showing that if  $\phi$  is unsatisfiable, then

$$\omega(G_\phi^{k \times \ell}) \leq \omega(G_\phi) + O\left( \sqrt{\frac{N}{k\ell}} \right) \leq 1 - \Omega(1), \quad (38)$$

provided we set  $k = \ell = c\sqrt{N}$  for a sufficiently large constant  $c$ . Theorem 26, in turn, gives us the following corollary.

**Corollary 28** *There exists an  $\text{AM}(2)$  protocol for 3SAT that uses  $\tilde{O}(\sqrt{n})$  communication, and that has a  $1$  vs.  $1 - \epsilon$  completeness/soundness gap for some constant  $\epsilon > 0$ .*

**Proof.** Given a 3SAT instance  $\varphi$  of size  $n$ , we apply Theorem 24 to get a PCP  $\phi$  of size  $N = n \text{ polylog } n$ . We then consider the birthday game  $G_\phi^{k \times k}$ , where  $k = c\sqrt{N}$  for some large constant  $c$ . Clearly, if  $\varphi$  is satisfiable then  $\omega(G_\phi^{k \times k}) = 1$ , while if  $\varphi$  is unsatisfiable then  $\omega(G_\phi^{k \times k}) \leq 1 - \epsilon$  for some constant  $\epsilon > 0$ . The only further observation we need to make is that Arthur can apply his verification function  $V_{BD}$  in time polynomial in  $n$ . ■

Of course, one way to state our  $\text{AM}(2)$  protocol is as a *reduction*: starting with a 3SAT instance  $\varphi$  of size  $n$ , we produce a free game  $G$  of size  $2^{\tilde{O}(\sqrt{n})}$  in  $2^{\tilde{O}(\sqrt{n})}$  time, such that  $\omega(G) = 1$  if  $\varphi$  is satisfiable and  $\omega(G) \leq 1 - \epsilon$  if  $\varphi$  is unsatisfiable. This immediately implies that, assuming the Exponential Time Hypothesis, there must be some constant  $\epsilon > 0$  such that the  $\text{FREEGAME}_\epsilon$  problem requires  $n^{\tilde{\Omega}(\log n)}$  time for all  $\epsilon \leq \epsilon$ .

However, we would like to do better than that, and also understand how the complexity of  $\text{FREEGAME}_\epsilon$  depends on the error  $\epsilon = \epsilon(n)$ . Unfortunately, our previous analysis was deficient in two ways: one that becomes relevant when  $\epsilon$  is very small, and another that becomes relevant when  $\epsilon$  is large. The first deficiency is that, while we showed that the distributions  $\mathcal{D}$  and  $\mathcal{U}$  had variation distance  $O(\sqrt{N/k\ell})$ , that bound gives nothing if  $k, \ell \ll \sqrt{N}$ , which is the relevant situation for small  $\epsilon$ . And this prevents us from showing that, if  $\epsilon = o(1)$ , then  $\text{FREEGAME}_\epsilon$  requires  $n^{\tilde{\Omega}(\epsilon^{-1} \log n)}$  time assuming the ETH. The second deficiency is that, because of our reliance on the clause/variable game, we were unable to prove *anything* when  $\epsilon$  was greater than some small, fixed constant  $\epsilon$ . This is particularly inconvenient, since it prevents us from saying that we have an “ $\text{AM}(2)$  protocol,” if  $\text{AM}(2)$  is defined with the conventional completeness/soundness gap of  $2/3$  vs.  $1/3$ . The next two subsections will remedy these deficiencies.

## 6.2 The High-Error Case

Our goal, in this subsection, is to show that if  $\epsilon = o(1)$ , then deciding whether  $\omega(G) = 1$  or  $\omega(G) \leq 1 - \epsilon$  for a given free game  $G$  requires  $n^{\tilde{\Omega}(\epsilon^{-1} \log n)}$  time assuming the ETH. (Later, Theorem 40 will give an algorithm that nearly achieves this lower bound.) To prove the  $\epsilon$ -dependent hardness result, we first need a simple combinatorial lemma, which can be seen as a generalization of the Birthday Paradox to regular bipartite graphs.

**Lemma 29** *Consider a bipartite graph, with  $M$  left-vertices each having degree  $c$ , and  $N$  right-vertices each having degree  $d$ . Choose  $k$  left-vertices and  $\ell$  right-vertices uniformly at random, and let  $H$  be the induced subgraph that they form. Then*

$$\Pr[H \text{ contains an edge}] \geq \frac{ck\ell}{N} \left(1 - \frac{c^2k^2}{N} - \frac{ck\ell}{N}\right). \quad (39)$$

**Proof.** Given a left-vertex  $v \in [M]$ , let  $\mathcal{N}(v) \subseteq [N]$  be the set of right-neighbors of  $v$ ; thus  $|\mathcal{N}(v)| = c$  for all  $v$ . Then by regularity, for any fixed  $w \in [N]$  we have

$$\Pr_{v \in [M]} [w \in \mathcal{N}(v)] = \frac{c}{N} \quad (40)$$

and

$$\Pr_{v, v' \in [M] : v \neq v'} [w \in \mathcal{N}(v) \cap \mathcal{N}(v')] \leq \left(\frac{c}{N}\right)^2. \quad (41)$$

Now let  $A$  be the set of left-vertices in  $H$  (thus  $|A| = k$ ), and let  $E$  denote the event that there exist two vertices  $v, v' \in A$  with a common neighbor. Then by the union bound,

$$\Pr[E] \leq \binom{k}{2} \sum_{w \in [N]} \Pr_{v, v' \in [M] : v \neq v'} [w \in \mathcal{N}(v) \cap \mathcal{N}(v')] \quad (42)$$

$$\leq \binom{k}{2} \cdot N \left(\frac{c}{N}\right)^2 \quad (43)$$

$$\leq \frac{c^2 k^2}{N}. \quad (44)$$

Furthermore, if  $E$  fails, then the left-vertices in  $H$  have  $ck$  distinct neighbors. So by the Bonferroni inequality, which states (as a special case) that

$$(1 - \varepsilon)^n \leq 1 - \varepsilon n + (\varepsilon n)^2, \quad (45)$$

we have

$$\Pr[H \text{ contains no edge} \mid \overline{E}] \leq \left(1 - \frac{ck}{N}\right)^\ell \leq 1 - \frac{ck\ell}{N} + \left(\frac{ck\ell}{N}\right)^2. \quad (46)$$

Hence

$$\Pr[H \text{ contains an edge}] \geq \left(1 - \frac{c^2 k^2}{N}\right) \left(\frac{ck\ell}{N} - \left(\frac{ck\ell}{N}\right)^2\right) \quad (47)$$

$$\geq \frac{ck\ell}{N} \left(1 - \frac{c^2 k^2}{N} - \frac{ck\ell}{N}\right). \quad (48)$$

■

We can now prove a more refined upper bound on  $\omega(G_\phi^{k \times \ell})$ , the success probability in the birthday game, in the case where  $k$  and  $\ell$  are small and  $\omega(G_\phi)$  is bounded away from 1.

**Lemma 30** *Suppose that  $\omega(G_\phi) \leq 1 - \epsilon$  and  $k, \ell \leq \sqrt{\epsilon N}/4$  for some absolute constant  $\epsilon > 0$ . Then*

$$\omega(G_\phi^{k \times \ell}) \leq 1 - \Omega\left(\frac{k\ell}{N}\right). \quad (49)$$

**Proof.** Reusing notation from Section 6.1 (and in particular, from the proof of Lemma 27), we have

$$\omega(G_\phi^{k \times \ell}) \leq \mathbb{E}_{\mathcal{U}}[V_{BD}] \quad (50)$$

$$= \sum_{I,J} \Pr_{\mathcal{U}}[I, J] \cdot V_{BD}(I, J, a(I), b(J)) \quad (51)$$

$$\leq \Pr_{\mathcal{U}}[S_{IJ} = 0] + \sum_{I,J:S_{IJ} \geq 1} \Pr_{\mathcal{U}}[I, J] \cdot V_{BD}(I, J, a(I), b(J)) \quad (52)$$

$$= \Pr_{\mathcal{U}}[S_{IJ} = 0] + \sum_{I,J:S_{IJ} \geq 1} \Pr_{\mathcal{D}}[(I, J)] \frac{3k\ell/N}{S_{IJ}} \cdot V_{BD}(I, J, a(I), b(J)) \quad (53)$$

$$\leq \Pr_{\mathcal{U}}[S_{IJ} = 0] + \frac{3k\ell}{N} \sum_{I,J:S_{IJ} \geq 1} \Pr_{\mathcal{D}}[(I, J)] \cdot V_{BD}(I, J, a(I), b(J)) \quad (54)$$

$$= \Pr_{\mathcal{U}}[S_{IJ} = 0] + \frac{3k\ell}{N} \mathbb{E}_{\mathcal{D}}[V_{BD}] \quad (55)$$

$$\leq \Pr_{\mathcal{U}}[S_{IJ} = 0] + \frac{3k\ell}{N} \omega(G_\phi) \quad (56)$$

$$\leq \left(1 - \frac{3k\ell}{N} \left(1 - \frac{9k^2}{N} - \frac{3k\ell}{N}\right)\right) + \frac{3k\ell}{N} (1 - \epsilon) \quad (57)$$

$$= 1 - \frac{3k\ell}{N} \left(\epsilon - \frac{9k^2}{N} - \frac{3k\ell}{N}\right) \quad (58)$$

$$= 1 - \Omega\left(\frac{k\ell}{N}\right), \quad (59)$$

where line (57) used Lemma 29, and line (59) used the assumption that  $k, \ell \leq \sqrt{\epsilon N}/4$ . ■

Lemma 30 has the following corollary, which gives a counterpart of Theorem 6 for AM[2] protocols with less than  $\sqrt{n}$  communication.

**Corollary 31** *For all  $\epsilon > 0$ , there exists an AM(2) protocol for 3SAT instances of size  $n$  which uses  $O(\sqrt{\epsilon n} \text{polylog } n)$  bits of communication, and which has a  $1$  vs.  $1 - \epsilon$  completeness/soundness gap.*

We also get the desired hardness result for FREEGAME.

**Theorem 32** *Assuming the ETH, there exists a constant  $\Delta > 0$  such that  $\text{FREEGAME}_\epsilon$  requires  $n^{\tilde{\Omega}(\epsilon^{-1} \log n)}$  deterministic time, for all  $\epsilon \in [1/n, \Delta]$ . (Likewise,  $\text{FREEGAME}_\epsilon$  requires  $n^{\tilde{\Omega}(\epsilon^{-1} \log n)}$  randomized time assuming the Randomized ETH.)*

**Proof.** Set  $\Delta := \epsilon/16$ , where  $\epsilon$  is the constant from Lemma 30. Fix a function  $\epsilon = \epsilon(M) \in [1/M, \Delta]$ , and suppose that  $\text{FREEGAME}_\epsilon$  instances of size  $M$  were solvable in time

$$M^{o\left(\frac{\epsilon^{-1} \log M}{\text{polylog}(\epsilon^{-1} \log M)}\right)}. \quad (60)$$

We need to show how, using that, we could decide a 3SAT instance  $\varphi$  of size  $n$  in time  $2^{o(n)}$ , thereby violating the ETH. The first step is to convert  $\varphi$  into a PCP  $\phi$  of size  $N = n \text{polylog } n$ . Next, we



generate the birthday repetition  $G_\phi^{k \times k}$ , where  $k = \sqrt{\varepsilon N}$ . (Here we use the assumption  $\varepsilon \geq 1/n$  to ensure that  $k \geq 1$ , and we use the assumption  $\varepsilon \leq \epsilon/16$  to ensure that  $k \leq \sqrt{\epsilon N}/4$ .) Note that the sizes of  $G_\phi^{k \times k}$ 's question and answer sets are  $M = 2^{k \log N} = N^k$ .

If  $\phi$  is satisfiable then  $\omega(G_\phi^{k \times k}) = 1$ , while by Lemma 30, if  $\phi$  is unsatisfiable then

$$\omega(G_\phi^{k \times k}) \leq 1 - \Omega\left(\frac{k^2}{N}\right) < 1 - 2\varepsilon. \quad (61)$$

So by approximating  $\omega(G_\phi^{k \times k})$  to within  $\pm\varepsilon$ , we can distinguish these cases and thereby decide whether  $\varphi$  was satisfiable. Using our hypothesized algorithm for  $\text{FREEGAME}_\varepsilon$ , this takes time

$$\exp\left(o\left(\frac{\varepsilon^{-1} \log^2 M}{\text{polylog}(\varepsilon^{-1} \log M)}\right)\right) = \exp\left(o\left(\frac{(k^2/N)^{-1} \cdot k^2 \log^2 N}{\text{polylog}((k^2/N)^{-1} \log N^k)}\right)\right) \quad (62)$$

$$= \exp\left(o\left(\frac{N \log^2 N}{(\log(N/k^2) + \log k + \log \log N)^R}\right)\right) \quad (63)$$

for some constant  $R$ . Note that if  $k$  is large, then  $\log k$  is  $\Omega(\log N)$ , while if  $k$  is small, then  $\log(N/k^2)$  is  $\Omega(\log N)$ . Therefore, provided  $R$  is large enough, the denominator will contain enough factors of  $\log N$  to clear all the  $\log N$  factors in the numerator, and our algorithm will have running time  $\exp(o(n))$ , giving the desired violation of the ETH. This reduction produces a deterministic algorithm if the  $\text{FREEGAME}$  algorithm was deterministic, or randomized if the  $\text{FREEGAME}$  algorithm was randomized. ■

We conjecture that the bound of Theorem 32 could be improved to  $n^{\tilde{\Omega}(\varepsilon^{-2} \log n)}$ , by considering free games  $G$  with  $\omega(G) \approx 1/2$  rather than  $\omega(G) \approx 1$ . This is a problem that we leave to future work.

### 6.3 The Low-Error Case

There is one obvious question that we haven't yet addressed: can we give an  $\text{AM}(2)$  protocol for 3SAT with *near-perfect* soundness? Or equivalently, given a free game  $G$  and some tiny  $\delta > 0$ , can we show that (assuming the ETH) there is no polynomial-time algorithm even to decide whether  $\omega(G) = 1$  or  $\omega(G) < \delta$ ? In this section we show that, using high-powered PCP machinery, we can indeed do this, although the result we get is probably not optimal.

One's first idea would be to apply ordinary parallel repetition to the birthday game—i.e., to consider  $(G_\phi^{k \times \ell})^m$  for some  $m > 1$ . Alas, this fails to work for an interesting reason. Namely, in the statement of the Parallel Repetition Theorem (Theorem 12), there is a  $1/\log|A||B|$  factor in the exponent, which is known to be necessary in general by a result of Feige and Verbitsky [18]. That factor immediately pushes the running time of our putative 3SAT algorithm above  $2^{O(n)}$ , preventing a contradiction with the ETH.

Note that Rao [32] proved that, for the special case of *projection games*, one can dramatically improve the Parallel Repetition Theorem, to show that  $\omega(G^t) \leq \omega(G)^{\Omega(t)}$  with no dependence on  $\log|A||B|$ . Here a projection game is a two-prover game  $G = (X, Y, A, B, \mathcal{D}, V)$  (not necessarily free) with  $V \in \{0, 1\}$  such that, for every  $(x, y) \in X \times Y$  in the support of  $\mathcal{D}$  and every  $a \in A$ , there is a unique  $b \in B$  such that  $V(x, y, a, b) = 1$ . Unfortunately, while the clause/variable game itself is a projection game, its birthday repetition is not.

Recently, Shaltiel [34] proved a “derandomized” version of the Parallel Repetition Theorem for the special case of free games. In particular, given a free game  $G = (X, Y, A, B, V)$  with  $V \in \{0, 1\}$ , Shaltiel constructs a new free game  $G_t = (X_t, Y_t, A_t, B_t, V_t)$ , which satisfies  $\omega(G_t) \leq \omega(G)^t$ , as well as  $\omega(G_t) = 1$  whenever  $\omega(G) = 1$ . Furthermore, the question sets  $X_t$  and  $Y_t$  in Shaltiel’s game have size at most  $(|X||Y||A||B|)^{O(t)}$ , which is perfect for our application. Unfortunately, the answer sets  $A_t$  and  $B_t$  have size  $\exp((t \log |X||Y||A||B|)^C)$  for some large constant  $C$ , and this once again prevents the desired contradiction with the ETH.

Finally, if we try to apply parallel repetition to the clause/variable game *before* applying birthday repetition, then the situation is even worse. For even one or two rounds of parallel repetition will blow up the question sets  $X$  and  $Y$  so that they no longer have size  $n^{1+o(1)}$ , meaning that we no longer have any hope of finding a collision using  $n^{1/2+o(1)}$  rounds of birthday repetition.

Currently, then, the best approach we know to the low-error case is simply to choose a PCP that *already* has low error, and then ensure that birthday repetition does not increase its error much further. In particular, rather than Theorem 24, we can start with the following result of Moshkovitz and Raz [31]:

**Theorem 33 (PCP Theorem, Moshkovitz-Raz Version [31])** *Given a 3SAT instance  $\varphi$  of size  $n$  as well as  $\delta > 0$ , it is possible in  $\text{poly}(n)$  time to produce a 2-CSP instance  $\phi$ , with  $n^{1+o(1)} \text{poly}(1/\delta)$  variables and constraints, and over an alphabet  $\Sigma$  of size  $|\Sigma| \leq 2^{\text{poly}(1/\delta)}$ , such that:*

- **(Completeness)** *If  $\text{SAT}(\varphi) = 1$  then  $\text{SAT}(\phi) = 1$ .*
- **(Soundness)** *If  $\text{SAT}(\varphi) < 1$  then  $\text{SAT}(\phi) < \delta$ .*
- **(Balance)** *The constraint graph of  $\phi$  is bipartite, and every variable appears in exactly  $d$  constraints, for some  $d = \text{poly}(1/\delta)$ .*

Since Theorem 33 outputs a 2-CSP  $\phi$ , we do not even need to consider the clause/variable game. Rather,  $\phi$  directly gives rise to a two-prover game  $H_\phi$ , in which Arthur chooses a constraint  $C$  of  $\phi$  uniformly at random, sends one of  $C$ ’s variables to Merlin<sub>1</sub> and the other to Merlin<sub>2</sub>, gets back their values, and accepts if and only if the values satisfy  $C$ . Clearly, if  $\text{SAT}(\varphi) = 1$  then  $\omega(H_\phi) = 1$ , while if  $\text{SAT}(\varphi) < 1$  then  $\omega(H_\phi) < \delta$ .

Now let  $N = n^{1+o(1)} \text{poly}(1/\delta)$  be the number of variables in  $\phi$ , let  $k, \ell \in [N]$ , and consider the birthday repetition  $H_\phi^{k \times \ell}$ . Observe that, in the variation distance argument from Section 6.1, the only special property of  $G_\phi$  that we used was the *regularity* of the constraint graph, and that property also holds for  $H_\phi$ . For this reason, we immediately get the following counterpart of Theorem 26:

**Theorem 34** *For all  $k, \ell \in [N]$ ,*

$$\omega(H_\phi) \geq \omega(H_\phi^{k \times \ell}) - O\left(\sqrt{\frac{N}{k\ell}}\right). \quad (64)$$

So in particular, suppose we set  $k := \sqrt{N}/\delta$ . Then if  $\text{SAT}(\varphi) < 1$ , we find that

$$\omega(H_\phi^{k \times k}) \leq \omega(H_\phi) + O(\sqrt{N}/k) = O(\delta). \quad (65)$$

Of course, if  $\text{SAT}(\varphi) = 1$  then  $\omega(H_\phi^{k \times k}) = 1$ . This gives us the following corollary:

**Corollary 35** *For all  $\delta > 0$ , there exists an  $\text{AM}(2)$  protocol for 3SAT instances of size  $n$  which uses  $n^{1/2+o(1)} \text{poly}(1/\delta)$  bits of communication, and which has a 1 vs.  $\delta$  completeness/soundness gap.*

We also get the following hardness result for distinguishing  $\omega(G) = 1$  from  $\omega(G) < \delta$ :

**Theorem 36** *Assuming the ETH, any deterministic algorithm to decide whether  $\omega(G) = 1$  or  $\omega(G) < \delta$ , given as input a description of a free game  $G$  of size  $n$ , requires  $n^{\text{poly}(\delta) \cdot (\log n)^{1-o(1)}}$  time. (Likewise, any randomized algorithm requires  $n^{\text{poly}(\delta) \cdot (\log n)^{1-o(1)}}$  time assuming the Randomized ETH.)*

**Proof.** Given a free game  $G$  of size  $M$ , suppose we could decide whether  $\omega(G) = 1$  or  $\omega(G) < \delta$  in time  $M^{p(\delta) \cdot (\log M)^{1-\eta}}$ , for some constant  $\eta > 0$  and sufficiently large polynomial  $p$ . We need to show how, using that, we could decide a 3SAT instance  $\varphi$  of size  $n$  in time  $2^{o(n)}$ , thereby violating the ETH. The first step is to convert  $\varphi$  into a 2-CSP  $\phi$  with  $N = n^{1+o(1)} \text{poly}(1/\delta)$  variables, using Theorem 33. Observe that the game  $H_\phi = (X, Y, A, B, V)$  satisfies  $|X| = |Y| = N$  and  $|A| = |B| = |\Sigma| = 2^{\text{poly}(1/\delta)}$ .

Next, we generate the birthday repetition  $H_\phi^{k \times k}$ , where  $k := c\sqrt{N}/\delta$  for some suitable constant  $c$ . Note that  $H_\phi^{k \times k}$  has question sets of size

$$N^k = \exp\left(\frac{c\sqrt{N} \log N}{\delta}\right) = \exp\left(\frac{n^{1/2+o(1)}}{\text{poly}(\delta)}\right) \quad (66)$$

and answer sets of size

$$2^k \text{poly}(1/\delta) = \exp\left(\frac{n^{1/2+o(1)}}{\text{poly}(\delta)}\right). \quad (67)$$

Thus, we set  $M := \exp(n^{1/2+o(1)}/\text{poly}(\delta))$ .

If  $\phi$  is satisfiable then  $\omega(H_\phi^{k \times k}) = 1$ , while if  $\phi$  is unsatisfiable then  $\omega(H_\phi^{k \times k}) \leq \delta$  by equation (65), provided the constant  $c$  was large enough. So by distinguishing these cases, we can decide whether  $\varphi$  was satisfiable. Using our hypothesized algorithm, this takes time

$$\exp(p(\delta) \cdot \log^{2-\eta} M) = \exp\left(p(\delta) \left(\frac{n^{1/2+o(1)}}{\text{poly}(\delta)}\right)^{2-\eta}\right) = \exp\left(n^{1-\eta/2+o(1)}\right), \quad (68)$$

provided the polynomial  $p$  was large enough. This gives us our desired violation of the ETH. ■

We conjecture that, assuming the ETH, distinguishing  $\omega(G) = 1$  from  $\omega(G) < \delta$  for a free game  $G$  should require  $n^{\Omega(\frac{\log n}{\log 1/\delta})}$  time for all  $\delta \leq 1/2$ , matching an upper bound that we will give in Theorem 40. A first step toward proving this conjecture would be to improve Theorem 33 (the result of Moshkovitz and Raz), so that it gave alphabet size  $|\Sigma| \leq \text{poly}(1/\delta)$  rather than  $|\Sigma| \leq 2^{\text{poly}(1/\delta)}$ . This is a well-known open problem. However, even if that problem were solved, one would *also* need a more refined analysis of birthday repetition, to eliminate the dependence of  $k$  on  $1/\delta$  in the proof of Theorem 36.

## 6.4 Complexity Consequences

Setting  $\delta := 1/3$ , Theorem 34 finally puts us in a position to say that

$$3\text{SAT} \in \text{AM}_{n^{1/2+o(1)}}(2), \quad (69)$$

where  $\text{AM}_{n^{1/2+o(1)}}(2)$  is defined with a  $2/3$  vs.  $1/3$  completeness/soundness gap, as in Definition 21. If we further combine this with a tight Cook-Levin Theorem (see, e.g., Turlakis [35]), showing that every language  $L \in \text{NTIME}[n]$  can be efficiently reduced to a set of 3SAT instances of size  $n \text{ polylog } n$ , then we get the following corollary:

**Corollary 37**  $\text{NTIME}[n] \subseteq \text{AM}_{n^{1/2+o(1)}}(2)$ .

Let us observe that Corollary 37 is non-relativizing.

**Proposition 38** *There exists an oracle  $A$  relative to which  $\text{NTIME}^A[n] \not\subseteq \text{AM}_{n^{1/2+o(1)}}^A(2)$ .*

**Proof Sketch.** For each  $n$ , the oracle  $A$  encodes a Boolean function  $A_n : \{0, 1\}^n \rightarrow \{0, 1\}$ , which is either identically 0 or else 1 on exactly one input. Let  $L_A$  be the unary language defined by  $0^n \in L_A$  if there exists an  $x \in \{0, 1\}^n$  such that  $A_n(x) = 1$ , and  $0^n \notin L_A$  otherwise. Then certainly  $L_A \in \text{NTIME}^A[n]$  for all  $A$ . On the other hand, using standard diagonalization techniques, it is not hard to construct  $A$  in such a way that  $L_A \notin \text{AM}_{n^{1/2+o(1)}}^A(2)$ —or even  $L_A \notin \text{AM}_{n/4}^A(2)$ . Intuitively, if the Merlins send only  $n/4$  bits each to Arthur (so  $n/2$  bits in total), then regardless of how those bits depend on their random challenges, with high probability Arthur will still need to query  $A$  on at least  $2^{n/2}$  inputs to confirm that  $0^n \in L_A$ . We omit the details, which are similar to those in the paper of Fortnow and Sipser [19]. ■

We leave as an open problem whether Corollary 37 is algebrizing in the sense of Aaronson and Wigderson [3].

## 7 Limitations of Multi-Prover AM

Our goal in this section is to prove that our 3SAT protocol is essentially optimal assuming the ETH, that  $\text{AM}(k) \subseteq \text{EXP}$  for all  $k = \text{poly}(n)$ , and that  $\text{AM}(k) = \text{AM}$  for all  $k = O(\log n)$ .

The section is organized as follows. First, in Section 7.1, we give a quasipolynomial-time algorithm for estimating the value of a 2-player free game. This algorithm implies that  $\text{AM}(2) \subseteq \text{EXP}$ , and even (with some more work) that  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ . The algorithm also implies that, if there exists an  $\text{AM}(2)$  protocol for 3SAT using  $o(\sqrt{n})$  communication, then 3SAT is solvable in  $2^{o(n)}$  time. In Section 7.2, we go further, using a result of Barak et al. [11] about subsampling dense CSPs to show that the value of any free game can be approximated by the value of a logarithmic-sized random subgame, and as a consequence, that  $\text{AM}(2) = \text{AM}$ . Finally, in Section 7.3, we generalize these results from  $\text{AM}(2)$  to  $\text{AM}(k)$  for all  $k = \text{poly}(n)$ .

### 7.1 The Basic Approximation Algorithm

We now explain how to approximate the value of a free game in quasipolynomial time.

**Theorem 39**  $\text{FREEGAME}_\varepsilon$  is solvable in time  $n^{O(\varepsilon^{-2} \log n)}$ . In more detail, given as input a description of a free game  $G = (X, Y, A, B, V)$ , there exists a randomized algorithm running in time  $|X| \cdot |A|^{O(\varepsilon^{-2} \log |Y| |B|)}$ , which estimates  $\omega(G)$  to within additive error  $\pm \varepsilon$ , with at least  $2/3$  success probability. There also exists a deterministic algorithm running in time  $(|X| |A|)^{O(\varepsilon^{-2} \log |Y| |B|)}$ .

**Proof.** The randomized estimation algorithm, call it **REst**, works as follows. First **REst** chooses a subset of questions  $S \subseteq X$  uniformly at random, subject to  $|S| = \kappa$  where

$$\kappa := \left\lceil \frac{\ln(6|Y||B|)}{\varepsilon^2} \right\rceil. \quad (70)$$

Next **REst** loops over all  $|A|^\kappa$  possible settings  $\alpha : S \rightarrow A$  of the answers to the  $\kappa$  questions in  $S$ . For each such  $\alpha$ , **REst** does the following:

- (1) It computes Merlin<sub>2</sub>'s "optimal response"  $b_\alpha : Y \rightarrow B$  to  $\alpha$ , supposing that Merlin<sub>1</sub> was only asked questions in  $S$ . For each question  $y \in Y$ , in other words, **Est** finds a response  $b_\alpha(y) \in B$  that maximizes

$$\mathbb{E}_{x \in S} [V(x, y, \alpha(x), b_\alpha(y))] \quad (71)$$

(breaking ties arbitrarily).

- (2) It computes Merlin<sub>1</sub>'s "optimal response"  $a_\alpha : X \rightarrow A$  to  $b_\alpha$ . For each  $x \in X$ , in other words, **REst** finds an  $a_\alpha(x) \in A$  that maximizes

$$\mathbb{E}_{y \in Y} [V(x, y, a_\alpha(x), b_\alpha(y))] \quad (72)$$

- (3) It computes the "value" obtained from the setting  $\alpha$ , as follows:

$$W_\alpha := \mathbb{E}_{x \in X, y \in Y} [V(x, y, a_\alpha(x), b_\alpha(y))] \quad (73)$$

Finally, **REst** computes

$$W := \max_{\alpha} W_\alpha, \quad (74)$$

and outputs  $W + \varepsilon$  as its estimate for  $\omega(G)$ .

Clearly **REst** runs in time

$$O(|A|^\kappa (|Y||B|\kappa + |X||A||Y| + |X||Y|)) = |X| \cdot |A|^{O(\varepsilon^{-2} \log |Y| |B|)}. \quad (75)$$

To prove correctness, we need to argue that, with high probability over the choice of  $S$ , we have

$$|(W + \varepsilon) - \omega(G)| \leq \varepsilon. \quad (76)$$

First observe that  $W_\alpha \leq \omega(G)$  for every  $\alpha$ . Therefore  $W \leq \omega(G)$  as well, and  $W + \varepsilon \leq \omega(G) + \varepsilon$ . So it suffices to prove the other direction: that  $W \geq \omega(G) - 2\varepsilon$  with at least  $2/3$  probability over  $S$ .

Let  $a^* : X \rightarrow A$  be an *optimal* strategy for Merlin<sub>1</sub> in the game  $G$ : that is, a strategy that, when combined with an optimal response  $b^* : Y \rightarrow B$  by Merlin<sub>2</sub>, achieves the value  $\omega(G)$ . Also,

fix a particular question  $y \in Y$  and answer  $b \in B$ . Then since the function  $V$  is  $[0, 1]$ -valued, Hoeffding's inequality (which also holds in the case of sampling without replacement) gives us

$$\Pr_{S \subseteq X, |S|=\kappa} \left[ \left| \mathbb{E}_{x \in S} [V(x, y, a^*(x), b)] - \mathbb{E}_{x \in X} [V(x, y, a^*(x), b)] \right| > \varepsilon \right] < 2e^{-\varepsilon^2 \kappa}. \quad (77)$$

So by the union bound, if we choose  $S$  randomly, then we have

$$\left| \mathbb{E}_{x \in S} [V(x, y, a^*(x), b)] - \mathbb{E}_{x \in X} [V(x, y, a^*(x), b)] \right| \leq \varepsilon \quad (78)$$

for every  $y \in Y$  and  $b \in B$  simultaneously, with probability at least

$$1 - 2e^{-\varepsilon^2 \kappa |Y| |B|} \geq \frac{2}{3} \quad (79)$$

over  $S$ . So suppose the inequality (78) holds. Let  $\alpha^* : S \rightarrow A$  be the restriction of the optimal strategy  $a^*$  to the set  $S$ , and let  $b_{\alpha^*} : Y \rightarrow B$  be an optimal response to  $\alpha^*$ . Then

$$W \geq W_{\alpha^*} \quad (80)$$

$$= \max_{a: X \rightarrow A} \mathbb{E}_{x \in X, y \in Y} [V(x, y, a(x), b_{\alpha^*}(y))] \quad (81)$$

$$\geq \mathbb{E}_{x \in X, y \in Y} [V(x, y, a^*(x), b_{\alpha^*}(y))] \quad (82)$$

$$\geq \mathbb{E}_{x \in S, y \in Y} [V(x, y, a^*(x), b_{\alpha^*}(y))] - \varepsilon \quad (83)$$

$$\geq \mathbb{E}_{x \in S, y \in Y} [V(x, y, a^*(x), b^*(y))] - \varepsilon \quad (84)$$

$$\geq \mathbb{E}_{x \in X, y \in Y} [V(x, y, a^*(x), b^*(y))] - 2\varepsilon \quad (85)$$

$$= \omega(G) - 2\varepsilon, \quad (86)$$

where lines (83) and (85) used inequality (78).

Finally, to get a deterministic estimation algorithm—call it **Est**—we simply need to loop over all possible  $S \subseteq X$  with  $|S| = \kappa$ , rather than choosing  $S$  randomly. We then output the maximum of  $W + \varepsilon$  over all  $S$  as our estimate for  $\omega(G)$ . This yields a running time of

$$|X|^\kappa \cdot |X| |A|^{O(\varepsilon^{-2} \log |Y| |B|)} = (|X| |A|)^{O(\varepsilon^{-2} \log |Y| |B|)}. \quad (87)$$

■

Let us point out some simple modifications to the algorithm **Est** from Theorem 39, which can improve its running time of  $n^{O(\varepsilon^{-2} \log n)}$  in certain cases.

**Theorem 40** *Given a free game  $G$  of size  $n$ , there is a deterministic algorithm running in  $n^{O(\varepsilon^{-1} \log n)}$  time to decide whether  $\omega(G) = 1$  or  $\omega(G) \leq 1 - \varepsilon$  (promised that one of those is the case), and there is a deterministic algorithm running in  $n^{O\left(1 + \frac{\log n}{\log 1/\delta}\right)}$  time to decide whether  $\omega(G) = 1$  or  $\omega(G) < \delta$ .*

**Proof.** In both cases, the key observation is that when running **Est**, we no longer need to estimate the quantity  $\mathbb{E}_{x \in X} [V(x, y, a^*(x), b)]$  to within  $\pm \varepsilon$ , and to pay the  $1/\varepsilon^2$  price that comes from Hoeffding’s inequality for doing so. Instead, for each  $y \in Y$  and  $b \in B$ , we simply need to know whether  $\mathbb{E}_{x \in X} [V(x, y, a^*(x), b)]$  is 1 or less than 1. Or equivalently, does there exist a “bad”  $x \in X$ —one such that  $V(x, y, a^*(x), b) < 1$ ? Moreover, we are promised that, if such a bad  $x$  *does* exist, then at least an  $\varepsilon$  or a  $1 - \delta$  fraction (respectively) of all  $x$ ’s are bad. Thus, when choosing the subset of questions  $S \subseteq X$ , it suffices to ensure that, with nonzero probability over  $S$ , we succeed in sampling one of the bad  $x$ ’s for every  $y \in Y$  and  $b \in B$ . By the union bound, this means that it suffices if, respectively,

$$(1 - \varepsilon)^\kappa < \frac{1}{3|Y||B|} \quad \text{or} \quad \delta^\kappa < \frac{1}{3|Y||B|} \quad (88)$$

where  $\kappa = |S|$ . Solving, we get respectively  $\kappa = O(\varepsilon^{-1} \log |Y||B|)$  or  $\kappa = O\left(1 + \frac{\log |Y||B|}{\log 1/\delta}\right)$ . Now we just need to plug the lower values of  $\kappa$  into equation (75) from the proof of Theorem 39 to get the improved running times. ■

The proof of Theorem 39 has a curious property. Namely, we showed that the value  $\omega(G)$  can in some sense be well-approximated by restricting attention to a random subset of questions  $S \subseteq X$  of logarithmic size. However, if  $G_S$  is the subgame obtained from  $G$  by restricting  $X$  to  $S$ , then the proof did *not* imply that  $\omega(G_S) \approx \omega(G)$ ! Using Hoeffding’s inequality, one can easily show that  $\omega(G_S) \geq \omega(G) - \varepsilon$  with high probability over the choice of  $S$ . The difficulty comes from the other direction—ironically, the “trivial” direction in the proof of Theorem 39. To get that  $W_\alpha \leq \omega(G)$ , we implicitly used the fact that  $W_\alpha$  was the value of a strategy pair  $(a_\alpha, b_\alpha)$  for the *whole* game  $G$ , not merely for the subgame  $G_S$ . Therefore, nothing we said implies that  $\omega(G_S) \leq \omega(G)$ , or even  $\omega(G_S) \leq \omega(G) + \varepsilon$ . And this makes intuitive sense: if the Merlins know that Merlin<sub>1</sub>’s question  $x$  will be restricted to a set of logarithmic size, then how do we know they can’t exploit that knowledge to win with greater probability? As we’ll discuss in Section 7.2, it turns out that one *can* prove the stronger result that  $\omega(G_S) \leq \omega(G) + \varepsilon$  with high probability over  $S$ —and this, in turn, lets one prove that  $\text{AM}(2) = \text{AM}$ . But more work (in particular, that of Alon et al. [5] and Barak et al. [11]) is needed.

Before we discuss that, let us point out some simple corollaries of Theorems 39 and 40.

**Corollary 41**  $\text{AM}(2) \subseteq \text{EXP}$ . *(In more detail, we can simulate any  $\text{AM}(2)$  protocol that uses  $p(n)$  communication and  $r(n) = \text{poly}(n)$  auxiliary randomness in  $2^{O(p(n)^2)+r(n)} \text{poly}(n)$  deterministic time, or  $2^{O(p(n)^2)} \text{poly}(n)$  randomized time.)*

**Proof.** Let  $L \in \text{AM}(2)$ . Then given an input  $x \in \{0, 1\}^n$ , the  $\text{AM}(2)$  protocol for checking whether  $x \in L$  can be represented as a free game  $G = (X, Y, A, B, V)$ , where  $X = Y = A = B = \{0, 1\}^{p(n)}$ , and where Arthur’s verification function  $V$  is computable in randomized  $\text{poly}(n)$  time using  $r(n)$  random bits. Now by Theorem 39, we can estimate  $\omega(G)$  to additive error (say)  $\pm 1/10$ , using  $(|X||A|)^{O(\log |Y||B|)} = 2^{O(p(n)^2)}$  deterministically-chosen evaluations of  $V$ . Furthermore, each of these  $V$  evaluations can be performed in  $\text{poly}(n)$  steps by a randomized algorithm (including the time needed for amplification to exponentially-small error probability), or in  $2^{r(n)} \text{poly}(n)$  steps by a deterministic algorithm. Finally, estimating  $\omega(G)$  lets us decide whether  $\omega(G) \geq 2/3$  or  $\omega(G) \leq 1/3$ , and hence whether  $x \in L$ . ■

Corollary 41 (and Theorem 40) have the following further consequence:

**Corollary 42** *If  $3\text{SAT} \in \text{AM}_{p(n)}(2)$ , then  $3\text{SAT} \in \text{TIME}[2^{O(p(n)^2)} \text{poly}(n)]$ . So in particular, assuming the Randomized ETH, any  $\text{AM}(2)$  protocol for  $3\text{SAT}$  with a  $1$  vs.  $1 - \varepsilon$  completeness/soundness gap must use  $\Omega(\sqrt{\varepsilon n})$  communication. Likewise, assuming the Randomized ETH, any protocol with a  $1$  vs.  $\delta$  gap must use  $\Omega(\sqrt{n \log 1/\delta})$  communication provided  $\delta \geq 2^{-n}$ . (If, moreover, Arthur’s verification procedure is deterministic, then it suffices to assume the ordinary ETH.)*

Also, a closer examination of the proof of Theorem 39 yields a better upper bound on  $\text{AM}(2)$  than  $\text{EXP}$ .

**Theorem 43**  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ .

**Proof Sketch.** We only sketch the proof, since in any case this result will be superseded by the later result that  $\text{AM}(2) = \text{AM}$ .

In the algorithm **REst** from Theorem 39, the first step has the form of an  $\text{AM}$  protocol. That is, following Corollary 41, let  $G = (X, Y, A, B, V)$  be the free game associated to the  $\text{AM}(2)$  protocol we want to simulate, with  $X = Y = A = B = \{0, 1\}^{\text{poly}(n)}$ . Then in our  $\text{AM}^{\text{NP}}$  simulation, we can have Arthur first choose a subset  $S \subseteq X$  of size  $\kappa = \text{poly}(n)$  uniformly at random and send  $S$  to Merlin. Next, using  $\kappa \log |A| = \text{poly}(n)$  bits, Merlin can send back a complete description of a response function  $\alpha : S \rightarrow A$  that is claimed to achieve (say)  $W_\alpha \geq 2/3$ . The question is how to implement the rest of the algorithm—or equivalently, how Arthur can verify that  $W_\alpha$  is large using an  $\text{NP}$  oracle.

Here the key idea is to use the property-testing paradigm of Goldreich, Goldwasser, and Ron [21]. As it stands, the inner loop of **REst** requires first computing Merlin<sub>2</sub>’s optimal response  $b_\alpha : Y \rightarrow B$  to  $\alpha$ , then computing Merlin<sub>1</sub>’s optimal response  $a_\alpha : X \rightarrow A$  to  $b_\alpha$ , and finally computing the value  $W_\alpha$  achieved by the pair  $(a_\alpha, b_\alpha)$ . In our case, all three of these steps would operate on  $2^{p(n)}$ -sized objects and require  $2^{O(p(n))}$  time.

However, by using a GGR-like approach, we can replace all three of these exponential-time computations by polynomial-time random sampling combined with  $\text{NP}$  oracle calls. In more detail: given  $\alpha$ , Arthur first chooses a subset  $T \subseteq Y$  of size  $\ell = \text{poly}(n)$  uniformly at random. He then uses his  $\text{NP}$  oracle to find a response function  $\beta : T \rightarrow B$  that maximizes

$$\mathbb{E}_{x \in S, y \in T} [V(x, y, \alpha(x), \beta(y))]. \quad (89)$$

Next, Arthur chooses *another* subset  $U \subseteq X$  of size  $m = \text{poly}(n)$  uniformly at random, and again uses his  $\text{NP}$  oracle to find a response function  $\gamma : U \rightarrow A$  that maximizes

$$W_\gamma := \mathbb{E}_{x \in U, y \in T} [V(x, y, \gamma(x), \beta(y))]. \quad (90)$$

Finally, Arthur accepts if and only if  $\max_\gamma W_\gamma \geq 1/2$ .

If  $\omega(G) \geq 2/3$ , then certainly Merlin can cause Arthur to accept with high probability in this protocol—for example, by sending Arthur  $\alpha = \alpha^*$ , the restriction of the globally optimal strategy  $a^* : X \rightarrow A$  to the subset  $S$ . As we saw in the proof of Theorem 39, the Hoeffding inequality and union bound ensure that  $\alpha^*$  “induces” responses  $\beta : T \rightarrow B$  by Merlin<sub>2</sub> that are close to the best possible responses in the full game  $G$ . Furthermore, even if  $T$  is only an  $O(\frac{1}{\varepsilon^2} \log(|X||A|))$ -sized subset of the full set  $Y$ , a *second* application of the Hoeffding inequality and union bound



ensure that  $\beta$ , in turn, induces responses  $\gamma : U \rightarrow A$  by Merlin<sub>1</sub> that are close to the best possible responses. So with high probability over the choices of  $S$ ,  $T$ , and  $U$ , the optimal response functions  $\beta : T \rightarrow B$  and  $\gamma : U \rightarrow A$  will achieve a value of  $W_\gamma$  close to  $\omega(G)$ .

As usual, the more interesting part is soundness: if  $\omega(G) \leq 1/3$ , then why can Merlin *not* cause Arthur to accept with high probability? The basic answer is that Merlin has to provide  $\alpha$  without knowing  $T$  or  $U$  (which Arthur will only choose later), and without being able to control  $\beta$  or  $\gamma$  (which are both just solutions to maximization problems, obtained using the NP oracle). As a consequence, one can show that, if  $\max_\gamma W_\gamma \geq 1/2$  with high probability over  $S$ ,  $T$ , and  $U$ , then one can construct a *global* strategy pair  $a : X \rightarrow A$ ,  $b : Y \rightarrow B$  that achieves value close to  $1/2$ . We omit the details, which closely follow those in the correctness proofs for property-testing algorithms due to Goldreich, Goldwasser, and Ron [21]. ■

## 7.2 Subsampling for Free Games and $\text{AM}(2) = \text{AM}$

In this section, we wish to go further than  $\text{AM}(2) \subseteq \text{EXP}$  or  $\text{AM}(2) \subseteq \text{AM}^{\text{NP}}$ , and prove that actually  $\text{AM}(2) = \text{AM}$ . For this purpose, given a free game  $G = (X, Y, A, B, V)$ , we need to show not merely that a near-optimal pair of strategies for  $G$  can be “induced” by examining a small random subgame  $G_S$ , but that  $\omega(G_S)$  *itself* gives a good approximation to  $\omega(G)$ , with high probability over  $S$ . As we explained in Section 7.1, it is easy to see that  $\mathbb{E}_S[\omega(G_S)] \geq \omega(G)$ , since we can start with optimal strategies for  $G$  and then restrict them to  $G_S$ . The hard part is to prove the other direction, that  $\mathbb{E}_S[\omega(G_S)] \leq \omega(G) + \varepsilon$ .

Here it is convenient to appeal to a powerful recent result of Barak et al. [11], which shows that *any* dense CSP over a finite alphabet  $\Sigma$  can be “subsampled,” generalizing an earlier subsampling result for the Boolean case by Alon et al. [5].

**Theorem 44 (Subsampling of Dense CSPs [11])** *Let  $\varphi$  be a  $k$ -CSP, involving  $n$  variables  $X = (x_1, \dots, x_n)$  over the finite alphabet  $\Sigma$ . Suppose that  $\varphi$  has “density”  $\alpha$ , in the following sense: for every collection  $Y \subset X$  of  $k-1$  variables,  $\varphi$  contains a  $[0, 1]$ -valued constraint  $C$  involving the variables  $Y \cup \{x_i\}$ , for at least an  $\alpha$  fraction of the remaining variables  $x_i \in X \setminus Y$ . Let  $\text{SAT}(\varphi) \in [0, 1]$  be the value of  $\varphi$ ; that is, the maximum of  $\mathbb{E}_{C \in \varphi}[C(X)]$  over all  $X \in \Sigma^n$ . Also, given a subset of variable indices  $I \subseteq [n]$ , let  $\varphi_I$  be the restriction of  $\varphi$  to the variables in  $I$  (and to those constraints that only involve  $I$  variables). Then provided we choose  $I$  uniformly at random subject to  $|I| \geq \frac{\log|\Sigma|}{\alpha\varepsilon^\Lambda}$  for some suitable constant  $\Lambda$ , we have*

$$\mathbb{E}_I[\text{SAT}(\varphi_I)] \leq \text{SAT}(\varphi) + \varepsilon. \quad (91)$$

As a side note, if the alphabet size  $|\Sigma|$  is constant, and if one does not care about the dependence of  $|I|$  on  $\varepsilon$ , then a version of Theorem 44 follows almost immediately from the Szemerédi Regularity Lemma, in its many-colored variant (see for example [29, Theorem 1.18]). However, this is of limited relevance to us, since in our case  $|\Sigma| = \text{poly}(n)$ .

We now use Theorem 44 to deduce an analogous subsampling theorem for free games.

**Theorem 45 (Subsampling of Free Games)** *Given a free game  $G = (X, Y, A, B, V)$  and  $\varepsilon > 0$ , let  $\kappa := 2\varepsilon^{-\Lambda} \log(|A| + |B|)$  (for some suitable constant  $\Lambda$ ), and assume  $\kappa \leq |X|$ . Choose a subset  $S \subseteq X$  of Merlin<sub>1</sub> questions uniformly at random subject to  $|S| = \kappa$ , and let  $G_S$  be the subgame of  $G$  with Merlin<sub>1</sub>’s questions restricted to  $S$ . Then*

$$\mathbb{E}_S[\omega(G_S)] \leq \omega(G) + \varepsilon. \quad (92)$$

**Proof.** We define a 2-CSP  $\varphi$  as follows. Let  $X' := X \times R_1$  and  $Y' := Y \times R_2$ , where  $R_1$  and  $R_2$  are finite sets chosen to ensure that  $|X||R_1| = |Y||R_2|$ . We think of  $X'$  and  $Y'$  as “augmented” versions of  $X$  and  $Y$  respectively, obtained by duplicating variables. Then  $\varphi$  will have a variable set consisting of  $a(x, r_1)$  for all  $(x, r_1) \in X'$  and  $b(y, r_2)$  for all  $(y, r_2) \in Y'$ , and alphabet  $\Sigma := A \cup B$ . For each  $(x, r_1) \in X'$  and  $(y, r_2) \in Y'$ , we will add a  $[0, 1]$ -valued constraint  $C$  between  $a(x, r_1)$  and  $b(y, r_2)$ , which behaves as follows:

- If  $a \in A$  and  $b \in B$ , then  $C(a, b) = V(x, y, a, b)$ .
- If  $a \notin A$  or  $b \notin B$ , then  $C(a, b) = 0$ .

In this way, we ensure the following two properties:

- (1)  $\text{SAT}(\varphi) = \omega(G)$ . Indeed, from any strategy pair  $(a, b)$  that achieves value  $\omega$  in  $G$ , we can construct an assignment to  $\varphi$  with value at least  $\omega$ , and conversely. (To see the converse, note that by convexity, if some  $a(x, r_1)$  takes on more than one value as we range over  $r_1 \in R_1$ , then there must be a single value  $a(x, r_1) = a(x)$  that does at least as well as the mean, and likewise for  $b(y, r_2)$ .)
- (2)  $\varphi$  has density  $\alpha = 1/2$  in the sense of Theorem 44. For it includes a constraint between every  $a$ -variable and every  $b$ -variable (although no constraints relating two  $a$ -variables or two  $b$ -variables), and the numbers of  $a$ -variables and  $b$ -variables are equal.

Now suppose we choose a subset  $I$  of the variables of  $\varphi$  uniformly at random, subject to  $|I| = \kappa$  where  $\kappa = 2\varepsilon^{-\Lambda} \log(|A| + |B|)$ . Then by Theorem 44, we have

$$\mathbb{E}_I[\text{SAT}(\varphi_I)] \leq \text{SAT}(\varphi) + \varepsilon = \omega(G) + \varepsilon. \quad (93)$$

To complete the proof, we need to show that

$$\mathbb{E}_S[\omega(G_S)] \leq \mathbb{E}_I[\text{SAT}(\varphi_I)]. \quad (94)$$

We will do so by appealing to the following general principle. Suppose we were to draw  $I$  by some random process in which we started with the set of all variables in  $\varphi$ , then repeatedly discarded variables until we were left with a uniformly-random subset  $I$  of size  $\kappa$ . Suppose, further, that at any point in this process, the distribution over constraints between remaining variables remained uniform over the set of *all* constraints  $C \in \varphi$ . Let  $J$  be a subset of variables obtained by stopping such a process at any intermediate point. Then we must have

$$\mathbb{E}_J[\text{SAT}(\varphi_J)] \leq \mathbb{E}_I[\text{SAT}(\varphi_I)]. \quad (95)$$

The reason is simply that, if we had a collection of partial assignments to the  $\varphi_J$ 's that achieved expected value  $\omega$ , then restricting those assignments to the  $\varphi_I$ 's would also achieve expected value  $\omega$ , by linearity of expectation.

So in particular, suppose we form  $J$  by choosing discarding all but  $\kappa$  variables of the form  $a(x, r_1)$ , (while keeping all variables of the form  $b(y, r_2)$ ). Then since the distribution over constraints remains uniform for this  $J$ , and since a sequence of further discardings could produce a uniformly-random subset  $I$  with  $|I| = \kappa$ , we have

$$\mathbb{E}_J[\text{SAT}(\varphi_J)] \leq \mathbb{E}_I[\text{SAT}(\varphi_I)] \leq \omega(G) + \varepsilon. \quad (96)$$

But  $E_J[\text{SAT}(\varphi_J)]$  is simply  $E_S[\omega(G_S)]$ , where  $S \subseteq X$  is a uniformly-random subset of Merlin<sub>1</sub> questions of size  $\kappa$ . This completes the proof. ■

Theorem 45 has the following easy corollary, which removes the “asymmetry” between the two Merlins.

**Corollary 46** *Given a free game  $G = (X, Y, A, B, V)$  and  $\varepsilon > 0$ , let  $\kappa := 2\varepsilon^{-\Lambda} \log(|A| + |B|)$ , and assume  $\kappa \leq \min\{|X|, |Y|\}$ . Choose  $S \subseteq X$  and  $T \subseteq Y$  uniformly at random and independently, subject to  $|S| = |T| = \kappa$ . Also, let  $G_{S,T}$  be the subgame of  $G$  with Merlin<sub>1</sub>’s questions restricted to  $S$  and Merlin<sub>2</sub>’s restricted to  $T$ . Then*

$$E_{S,T}[\omega(G_{S,T})] \leq \omega(G) + 2\varepsilon. \quad (97)$$

**Proof.** We simply need to apply Theorem 45 twice in succession, once to reduce Merlin<sub>1</sub>’s question set, and then a second time to reduce Merlin<sub>2</sub>’s. The result follows by linearity of expectation. ■

Using Corollary 46, we now prove that  $\text{AM}(2) = \text{AM}$ .

**Theorem 47**  $\text{AM}(2) = \text{AM}$ .

**Proof.** Let  $L \in \text{AM}(2)$ . Then just like in Corollary 41, an  $\text{AM}(2)$  protocol for checking whether a string is in  $L$  can be represented as a free game  $G = (X, Y, A, B, V)$ , where  $X = Y = A = B = \{0, 1\}^{p(n)}$  for some polynomial  $p$ , and  $V$  is computable in randomized  $\text{poly}(n)$  time.

Let  $\varepsilon := 1/24$  and  $\kappa := 2\varepsilon^{-\Lambda}(p(n) + 1)$ , and suppose we choose  $S \subseteq X$  and  $T \subseteq Y$  uniformly at random subject to  $|S| = |T| = \kappa$ . Then by Corollary 46,

$$E_{S,T}[\omega(G_{S,T})] \leq \omega(G) + \frac{1}{12}. \quad (98)$$

But this immediately gives us our  $\text{AM}$  simulation, as follows. First Arthur chooses  $S, T \subseteq \{0, 1\}^{p(n)}$  uniformly at random, subject to  $|S| = |T| = \kappa$  as above. He then sends  $S$  and  $T$  to Merlin, using  $2\kappa \cdot p(n) = O(p(n)^2)$  bits. Next Merlin replies with a pair of strategies  $a : S \rightarrow A$  and  $b : T \rightarrow B$ , which again takes  $O(p(n)^2)$  bits. Let

$$\omega_{S,T} := E_{x \in S, y \in T}[V(x, y, a(x), b(y))] \quad (99)$$

be the subsampled success probability; notice that  $\omega_{S,T} \leq \omega(G_{S,T})$  for all  $S, T$ . Then finally, if  $V$  is deterministic, then Arthur simply computes  $\omega_{S,T}$  and accepts if and only if  $\omega_{S,T} \geq 1/2$ . If  $V$  is randomized, then Arthur instead computes an estimate  $\tilde{\omega}_{S,T}$  such that

$$\Pr[|\tilde{\omega}_{S,T} - \omega_{S,T}| > 0.01] \leq \exp(-\kappa), \quad (100)$$

and accepts if and only if  $\tilde{\omega}_{S,T} \geq 0.51$ .

We claim, first, that this protocol has completeness error at most  $\exp(-\kappa)$ . For we can always consider an optimal pair of strategies  $a : X \rightarrow A$  and  $b : Y \rightarrow B$  for the full protocol, which achieve value

$$\omega := E_{x \in X, y \in Y}[V(x, y, a(x), b(y))] = E_{S,T}[\omega_{S,T}] \geq \frac{2}{3} \quad (101)$$

by assumption. Then a standard Chernoff bound implies that  $\omega_{S,T} \geq 0.52$ , and hence  $\tilde{\omega}_{S,T} \geq 0.51$ , with at least  $1 - \exp(-\kappa)$  probability over the choice of  $S$  and  $T$ .

We next upper-bound the soundness error. Suppose  $\omega(G) \leq 1/3$ ; then by equation (98),

$$\mathbb{E}_{S,T}[\omega_{S,T}] \leq \mathbb{E}_{S,T}[\omega(G_{S,T})] \leq \omega(G) + \frac{1}{12} \leq \frac{5}{12}. \quad (102)$$

So by Markov's inequality,

$$\Pr_{S,T} \left[ \omega_{S,T} \geq \frac{1}{2} \right] \leq \frac{5/12}{1/2} = \frac{5}{6}, \quad (103)$$

and hence

$$\Pr[\tilde{\omega}_{S,T} \geq 0.51] \leq \frac{5}{6} + \exp(-\kappa) \quad (104)$$

as well. So Arthur rejects with constant probability. Of course, we can amplify the completeness/soundness gap further by repeating the protocol. ■

### 7.3 The $k$ -Merlin Case

In this section, we generalize our results from  $\text{AM}(2)$  to  $\text{AM}(k)$  for larger  $k$ . The first step is to generalize Theorem 39, to obtain a nontrivial approximation algorithm for  $k$ -player free games.

**Theorem 48** *Let  $G$  be a  $k$ -player free game, with question sets  $Y_1, \dots, Y_k$  and answer sets  $B_1, \dots, B_k$  (assume  $|B_i| \geq 2$  for all  $i \in [k]$ ). There exists a deterministic algorithm that approximates  $\omega(G)$  to within additive error  $\pm \varepsilon$ , in time*

$$\exp \left( \frac{k^2}{\varepsilon^2} \sum_{i < j} \log(|Y_i| |B_i|) \cdot \log(|Y_j| |B_j|) \right) = n^{O(\varepsilon^{-2} k^2 \log n)}, \quad (105)$$

where  $n = |Y_1| |B_1| \cdots |Y_k| |B_k|$  is the input size.

**Proof.** The basic idea is to use a recursive generalization, call it  $\text{Est}_k$ , of the (deterministic) approximation algorithm  $\text{Est}$  from Theorem 39. The recursive version will “peel off the Merlins one at a time.” That is, given a description of a  $k$ -player free game  $G$  as input,  $\text{Est}_k$  will reduce the estimation of  $\omega(G)$  to the estimation of  $\omega(G')$ , for a quasipolynomial number of  $(k-1)$ -player free games  $G'$ , each one involving  $\text{Merlin}_1$  through  $\text{Merlin}_{k-1}$  only ( $\text{Merlin}_k$ 's behavior having already been fixed). Each  $\omega(G')$  will in turn be estimated by calling  $\text{Est}_{k-1}$ , and so on until  $k=1$ , at which point we can just do a straightforward maximization.

In more detail, let  $\delta := \varepsilon/k$ . Then for each  $\ell \in \{2, \dots, k\}$ , let

$$\kappa_\ell := \frac{C}{\delta^2} \sum_{i=1}^{\ell-1} \log(|Y_i| |B_i|), \quad (106)$$

for some suitable constant  $C$ . Then  $\text{Est}_k$  loops over all  $\binom{|Y_k|}{\kappa_k}$  subsets of questions  $S_k \subseteq Y_k$  such that  $|S_k| = \kappa_k$ , as well as all  $|B_k|^{\kappa_k}$  possible settings  $\alpha_k : S_k \rightarrow B_k$  of the answers to the  $\kappa_k$  questions in  $S_k$ . For each such pair  $P = (S_k, \alpha_k)$ , we define a  $(k-1)$ -player subgame  $G_P$ , which is played by  $\text{Merlin}_1$  through  $\text{Merlin}_{k-1}$ , and which has question sets  $Y_1, \dots, Y_{k-1}$  and answer sets  $B_1, \dots, B_{k-1}$ . The verification function of  $G_P$  is defined as follows:

$$V_P(y_1, \dots, y_{k-1}, b_1, \dots, b_{k-1}) := \mathbb{E}_{y_k \in S_k} [V(y_1, \dots, y_k, b_1, \dots, b_{k-1}, \alpha_k(y_k))]. \quad (107)$$

In other words,  $G_P$  is the same game as  $G$ , except that we assume that Merlin $_k$  is only asked questions  $y_k \in S_k$ , and that he responds to each with  $\alpha_k(y_k)$ .

Now, for each  $P$ , the algorithm **Est** $_k$  does the following:

- (1) If  $k \geq 3$ , then it calls **Est** $_{k-1}$  recursively, in order to find approximately optimal strategies  $(b_{P,i} : Y_i \rightarrow B_i)_{i \in [k-1]}$  for Merlin $_1$  through Merlin $_{k-1}$  in  $G_P$ . Here “approximately optimal” means achieving value at least  $\omega(G_P) - \delta$ . Of course, when  $k = 2$ , the algorithm can simply compute Merlin $_1$ ’s exactly-optimal response  $b_{P,1} : Y_1 \rightarrow B_1$  by maximizing

$$\mathbb{E}_{y_2 \in S_2} [V(y_1, y_2, b_{P,1}(y_1), \alpha_2(y_2))] \quad (108)$$

for each  $y_1 \in Y_1$  separately, just like in the two-player algorithm **Est**.

- (2) Given the responses  $b_{P,1}, \dots, b_{P,k-1}$  of Merlin $_1$  through Merlin $_{k-1}$ , the algorithm computes Merlin $_k$ ’s best response  $b_{P,k} : Y_k \rightarrow B_k$  on the full set  $Y_k$  by maximizing

$$\mathbb{E}_{y_1 \in Y_1, \dots, y_{k-1} \in Y_{k-1}} [V(y_1, \dots, y_k, b_{P,1}(y_1), \dots, b_{P,k}(y_k))] \quad (109)$$

for each  $y_k \in Y_k$  separately. It then lets

$$W_P := \mathbb{E}_{y_1 \in Y_1, \dots, y_k \in Y_k} [V(y_1, \dots, y_k, b_{P,1}(y_1), \dots, b_{P,k}(y_k))] \quad (110)$$

be the value of the  $k$ -tuple of strategies induced by  $P$ .

Finally, **Est** $_k$  outputs  $W := \max_P W_P$  as its estimate for  $\omega(G)$ . Note that, in addition to  $W$ , the algorithm also outputs a strategy  $k$ -tuple  $(b_{P,1}, \dots, b_{P,k})$  that achieves value  $W$ .

Let  $T(\ell)$  be the number of evaluations of the “original” verification function  $V(y_1, \dots, y_k, b_1, \dots, b_k)$  that **Est** $_\ell$  needs to make, when it’s called on an  $\ell$ -player game involving Merlin $_1$  through Merlin $_\ell$ . Then we have the following recurrence relation:

$$T(\ell) \leq \binom{|Y_\ell|}{\kappa_\ell} |B_\ell|^{\kappa_\ell} (T(\ell-1) + |Y_1| \cdots |Y_{\ell-1}| \cdot |Y_\ell| |B_\ell| \cdot \kappa_{\ell+1} \cdots \kappa_k), \quad (111)$$

with base case  $T(1) = |Y_1| |B_1| \cdot \kappa_2 \cdots \kappa_k$ . (The reason for the factor of  $\kappa_{\ell+1} \cdots \kappa_k$  is that, just to compute  $V$  for a game involving Merlin $_1$  through Merlin $_\ell$ , one needs to take an expectation over all  $y_{\ell+1} \in S_{\ell+1}, \dots, y_k \in S_k$ .) Now, it is not hard to see that the

$$|Y_1| \cdots |Y_{\ell-1}| \cdot |Y_\ell| |B_\ell| \cdot \kappa_{\ell+1} \cdots \kappa_k \quad (112)$$

terms all get absorbed by asymptotically larger terms. Asymptotically, then,

$$T(k) \leq (|Y_k| |B_k|)^{\kappa_k} T(k-1) \quad (113)$$

$$= \exp \left( \log(|Y_k| |B_k|) \cdot \frac{C}{\delta^2} \sum_{i=1}^{k-1} \log(|Y_i| |B_i|) \right) \cdot T(k-1) \quad (114)$$

$$= \exp \left( \frac{k^2}{\varepsilon^2} \sum_{i < j} \log(|Y_i| |B_i|) \cdot \log(|Y_j| |B_j|) \right). \quad (115)$$

Since the running time is dominated by evaluations of  $V$  (each of which takes constant time), this also gives the asymptotic running time.

The proof of correctness for  $\mathbf{Est}_k$  follows the same general outline as the proof of the correctness for  $\mathbf{Est}$ . Once again, since each  $W_P$  is the value achieved by some actual  $k$ -tuple of strategies  $b_{P,1}, \dots, b_{P,k}$  in the full game  $G$ , it is clear that  $W_P \leq \omega(G)$  for all  $P$ . The nontrivial part is to show that  $W_P \geq \omega(G) - \varepsilon$  for *some*  $P = (S_k, \alpha_k)$ .

We will prove this claim by induction on  $\ell$ . That is, suppose by induction that, for every  $(\ell - 1)$ -player game  $G_P$  played by Merlin<sub>1</sub> through Merlin <sub>$\ell-1$</sub> , the algorithm  $\mathbf{Est}_{\ell-1}$  finds an  $(\ell - 1)$ -tuple of strategies that achieve a value at least  $\omega(G_P) - \epsilon$ . We will show that this implies that, for every  $\ell$ -player game  $G_Q$  played by Merlin<sub>1</sub> through Merlin <sub>$\ell$</sub> , the algorithm  $\mathbf{Est}_\ell$  achieves a value at least  $\omega(G_Q) - \epsilon - \delta$ . Since  $\delta = \varepsilon/k$ , clearly this suffices to prove the claim.

Let  $G_Q$  be the  $\ell$ -player game defined by the tuple  $Q = (S_{\ell+1}, \dots, S_k, \alpha_{\ell+1}, \dots, \alpha_k)$ . Then  $G_Q$  has the verification function

$$V_Q(y_1, \dots, y_\ell, b_1, \dots, b_\ell) := \mathbb{E}_{y_{\ell+1} \in S_{\ell+1}, \dots, y_k \in S_k} [V(y_1, \dots, y_k, b_1, \dots, b_\ell, \alpha_{\ell+1}(y_{\ell+1}), \dots, \alpha_k(y_k))]. \quad (116)$$

By definition, there exists an  $\ell$ -tuple of strategies  $(b_i^* : Y_i \rightarrow B_i)_{i \in [\ell]}$  such that

$$\mathbb{E}_{y_1 \in Y_1, \dots, y_\ell \in S_\ell} [V_Q(y_1, \dots, y_\ell, b_1^*(y_1), \dots, b_\ell^*(y_\ell))] = \omega(G_Q). \quad (117)$$

Given a subset  $S_\ell \subseteq Y_\ell$  with  $|S_\ell| = \kappa_\ell$ , call  $S_\ell$  “good” if it has the property that

$$\left| \mathbb{E}_{y_\ell \in S_\ell} [V_Q(y_1, \dots, y_\ell, b_1, \dots, b_{\ell-1}, b_\ell^*(y_\ell))] - \mathbb{E}_{y_\ell \in Y_\ell} [V_Q(y_1, \dots, y_\ell, b_1, \dots, b_{\ell-1}, b_\ell^*(y_\ell))] \right| \leq \frac{\delta}{2} \quad (118)$$

for *every*  $(\ell - 1)$ -tuple of questions  $(y_1, \dots, y_{\ell-1}) \in Y_1 \times \dots \times Y_{\ell-1}$  and answers  $(b_1, \dots, b_{\ell-1}) \in B_1 \times \dots \times B_{\ell-1}$ . Then a straightforward application of the Hoeffding inequality and union bound shows that the fraction of  $S_\ell$ ’s that are good is at least

$$1 - 2e^{-\delta^2 \kappa_\ell} |Y_1| |B_1| \dots |Y_{\ell-1}| |B_{\ell-1}| = 1 - 2 \exp \left( -C \sum_{i=1}^{\ell-1} \log(|Y_i| |B_i|) \right) |Y_1| |B_1| \dots |Y_{\ell-1}| |B_{\ell-1}| \geq \frac{2}{3} \quad (119)$$

for suitable  $C$ . Thus, certainly there *exists* a good  $S_\ell$ , and  $\mathbf{Est}_\ell$  will find one when it loops over all possibilities. Fix a good  $S_\ell$  in what follows.

Let  $G_P$  be the  $(\ell - 1)$ -player game played by Merlin<sub>1</sub> through Merlin <sub>$\ell-1$</sub> , which is obtained from  $G_Q$  by restricting Merlin <sub>$\ell$</sub> ’s question set to  $S_\ell$ , and restricting Merlin <sub>$\ell$</sub> ’s strategy to  $b_\ell^*$ . Then notice that  $S_\ell$  being good has the following two consequences:

- (i) We can achieve value at least  $\omega(G_Q) - \delta/2$  in  $G_P$ , by simply starting with  $b_1^*, \dots, b_\ell^*$  and then restricting  $b_\ell^*$  to  $S_\ell$ .
- (ii) Any time we find strategies  $b_1, \dots, b_{\ell-1}$  that achieve value at least  $W$  in  $G_P$ , we have also found strategies that achieve value at least  $W - \delta/2$  in  $G_Q$ : we simply need to fix Merlin <sub>$\ell$</sub> ’s strategy to be  $b_\ell^*$ .

Combining facts (i) and (ii), we find that, if  $\mathbf{Est}_{\ell-1}$  can achieve value at least  $\omega(G_P) - \epsilon$  in  $G_P$ , then  $\mathbf{Est}_\ell$  can achieve value at least  $\omega(G_Q) - \epsilon - \delta$  in  $G_Q$ . Intuitively, this is because the

errors build up linearly: we incur an error of  $\delta/2$  when switching from  $G_Q$  to  $G_P$ , then an error of  $\epsilon$  (by hypothesis) when running  $\text{Est}_{\ell-1}$  to find strategies for  $G_P$ , and finally another error of  $\delta/2$  when switching from  $G_P$  back to  $G_Q$ . This completes the induction, and hence the proof that  $W \geq \omega(G) - \epsilon$ . ■

Just like in the  $k = 2$  case, we can modify the algorithm  $\text{Est}_k$  so that it chooses the sets  $S$  uniformly at random, rather than looping over all possible  $S$ 's. By doing so, we can get a randomized algorithm that approximates  $\omega(G)$  to within additive error  $\pm\epsilon$  in the slightly better running time

$$|Y_k| \cdot \exp \left( \frac{k^2}{\epsilon^2} \sum_{i < j} \log(|Y_i| |B_i|) \cdot \log(|B_j|) \right). \quad (120)$$

We omit the details.

Analogously to Theorem 40, we can also improve the running time of  $\text{Est}_k$  in the case of perfect completeness.

**Theorem 49** *Given a  $k$ -player free game  $G = (Y_1, \dots, Y_k, B_1, \dots, B_k, V)$ , we can decide whether  $\omega(G) = 1$  or  $\omega(G) < 1 - \epsilon$  (promised that one of those is the case) using a deterministic algorithm that runs in time  $n^{O(\epsilon^{-1} k^2 \log n)}$ , where  $n = |Y_1| |B_1| \cdots |Y_k| |B_k|$  is the input size. (In more detail, in both running time bounds of Theorem 48, we can improve the factor of  $k^2/\epsilon^2$  in the exponent to  $k^2/\epsilon$ .)*

**Proof Sketch.** As in Theorem 40, the key observation is that, if we only care about distinguishing  $\omega(G) = 1$  from  $\omega(G) < 1 - \epsilon$ , then it suffices to set

$$\kappa_\ell := \frac{C}{\epsilon/k^2} \sum_{i=1}^{\ell-1} \log(|Y_i| |B_i|). \quad (121)$$

The reason is this: we still need to limit the new error introduced at each level of the recursion to  $\delta = \epsilon/k$ . However, if  $\omega(G) = 1$ , then the total error will *never* exceed  $k(\epsilon/k) = \epsilon$ , given optimal responses to the question sets  $S_2, \dots, S_k$  chosen at each level of the recursion, assuming that  $S_2, \dots, S_k$  are good. And it is known that, if a  $[0, 1]$  random variable has expectation at most  $\epsilon$ , then we can estimate it to within additive error  $\pm\delta$  with high probability using only  $O(\epsilon/\delta^2)$  samples (see for example [1, Appendix 6]). The improved running time bounds follow directly from the improvement to  $\kappa_\ell$ . ■

Theorem 48 readily implies an upper bound on  $\text{AM}(k)$ .

**Corollary 50**  $\text{AM}(k) \subseteq \text{EXP}$  for all polynomials  $k = k(n)$ .

**Proof.** Let  $L \in \text{AM}(k)$ . Then given an input  $x \in \{0, 1\}^n$ , the  $\text{AM}(k)$  protocol for checking whether  $x \in L$  can be represented as a  $k$ -player free game  $G = ((Y_i)_{i \in [k]}, (B_i)_{i \in [k]}, V)$ , where  $Y_i = B_i = \{0, 1\}^{p(n)}$  for all  $i$  (for some polynomial  $p$ ), and where Arthur's verification function  $V$  is computable in  $\text{poly}(n)$  time using  $r(n) = \text{poly}(n)$  bits of randomness. Now by Theorem 48, we can estimate  $\omega(G)$  to additive error (say)  $\epsilon = 1/10$  by a deterministic algorithm that makes

$$\exp \left( \frac{k^2}{\epsilon^2} \sum_{i < j} p(n)^2 \right) = \exp \left( k^4 p(n)^2 \right) \quad (122)$$

evaluations of  $V$ . Furthermore, each  $V$  evaluation can be performed in deterministic time  $2^{r(n)} \text{poly}(n)$  (or in randomized time  $\text{poly}(n)$ , even allowing for amplification to exponentially small error probability). But this lets us decide whether  $\omega(G) \geq 2/3$  or  $\omega(G) \leq 1/3$ , and hence whether  $x \in L$ . ■

A second corollary of Theorem 48 is that, assuming the ETH, there is a hard  $\Omega(n^{1/4})$  limit on the amount of communication needed in any constant-soundness  $\text{AM}(k)$  protocol for 3SAT, regardless of  $k = k(n)$ . Furthermore, if  $k = n^{o(1)}$ , then  $n^{1/2-o(1)}$  communication is needed. (Later, in Section 7.4, we will improve this to show that  $\Omega(\sqrt{n})$  communication is needed regardless of  $k$ .)

**Corollary 51** *Assuming the Randomized ETH, any  $\text{AM}(k)$  protocol for 3SAT with a  $1$  vs.  $1 - \varepsilon$  completeness/soundness gap must use  $\Omega(k + \sqrt{\varepsilon n}/k) = \Omega((\varepsilon n)^{1/4})$  bits of communication in total. (Also, if Arthur’s verification procedure is deterministic, then it suffices to assume the ordinary ETH.)*

**Proof.** Assume for simplicity that  $\varepsilon = 1/2$ . Consider an  $\text{AM}(k)$  protocol that uses  $q(n)$  bits of communication in total. We can assume  $q(n) \geq k$ , since otherwise we could eliminate one of the Merlins and reduce to the  $\text{AM}(k-1)$  case. Now suppose that for all  $i \in [k]$ , Arthur sends an  $s_i$ -bit message to Merlin <sub>$i$</sub>  and receives a  $t_i$ -bit response. Then by Theorem 48, we can simulate the protocol to within constant error by an algorithm that makes

$$\exp \left( k^2 \sum_{i < j} (s_i + t_i)(s_j + t_j) \right) \leq \exp \left( \frac{k^2}{2} \left( \sum_{i=1}^k (s_i + t_i) \right)^2 \right) \leq \exp(k^2 q(n)^2) \quad (123)$$

evaluations of Arthur’s verification procedure  $V$ . Furthermore, each  $V$  evaluation can be performed in randomized  $\text{poly}(n)$  time (even allowing for amplification to exponentially small error probability). So if 3SAT requires  $2^{\Omega(n)}$  randomized time, then  $k^2 q(n)^2 = \Omega(n)$  and  $q(n) = \Omega(\sqrt{n}/k)$ . Combining with  $q(n) \geq k$  then yields  $q(n) = \Omega(n^{1/4})$ .

For general  $\varepsilon > 0$ , we simply need to use Theorem 49 rather than Theorem 48. For the last part, we note that if  $V$  is deterministic then so is our 3SAT algorithm. ■

## 7.4 Subsampling with $k$ Merlins

Finally, let us show that  $\text{AM}(k) = \text{AM}$  for all  $k = \text{poly}(n)$ . The first step is to generalize Theorem 45, the subsampling theorem for 2-player free games, to  $k$  players for arbitrary  $k$ . For technical reasons—related to the definition of “denseness” in the statement of Theorem 44—doing this will require reducing a free game to a  $k$ -CSP in a different way than we did in the proof of Theorem 45.<sup>13</sup>

**Theorem 52 (Subsampling of  $k$ -Player Free Games)** *Given a  $k$ -player free game*

$$G = (Y_1, \dots, Y_k, B_1, \dots, B_k, V) \quad (124)$$

<sup>13</sup>In more detail, suppose we tried to encode a  $k$ -player free game  $G$  as a  $k$ -CSP in the “obvious” way. Then among all possible  $k$ -tuples of variables, the fraction that were related by a nontrivial constraint would decrease like  $k!/k^k \approx e^{-k}$ , simply because any such  $k$ -tuple must involve exactly one variable for each of the  $k$  players, with no “collisions.” But this, in turn, would mean that we could only get the conclusion  $\text{AM}(k) = \text{AM}$  when  $k = O(\log n)$ : for larger  $k$ , our  $k$ -CSP simply wouldn’t be “dense” enough for Theorem 44 to give what we want. To get around this problem, we use a different encoding of  $G$  as a  $k$ -CSP: one in which every variable, individually, involves questions to all  $k$  of the players.



and  $\varepsilon > 0$ , let  $\kappa := \varepsilon^{-\Lambda} \log(|B_1| \cdots |B_k|)$  (for some suitable constant  $\Lambda$ ), and assume  $\kappa \leq \min\{|Y_1|, \dots, |Y_k|\}$ . For each  $i \in [k]$ , choose a subset  $S_i \subseteq Y_i$  of Merlin <sub>$i$</sub>  questions uniformly at random subject to  $|S_i| = \kappa$ , let  $S := S_1 \times \cdots \times S_k$ , and let  $G_S$  be the subgame of  $G$  with Merlin <sub>$i$</sub> 's questions restricted to  $S_i$ . Then

$$\mathbb{E}_S[\omega(G_S)] \leq \omega(G) + \varepsilon. \quad (125)$$

**Proof.** We define a  $k$ -CSP  $\varphi$  as follows. Let

$$\mathbf{Y} := Y_1 \times \cdots \times Y_k, \quad (126)$$

$$\mathbf{B} := B_1 \times \cdots \times B_k. \quad (127)$$

Then there is one variable, of the form  $\mathbf{b}(\mathbf{y}) \in \mathbf{B}$ , for every  $k$ -tuple  $\mathbf{y} \in \mathbf{Y}$ . (Thus, an assignment  $\mathbf{b} : \mathbf{Y} \rightarrow \mathbf{B}$  to  $\varphi$  will be a fairly large object, mapping  $k$ -tuples of questions to  $k$ -tuples of answers.) There is also a  $[0, 1]$ -valued constraint,  $C_{\mathbf{R}}$ , for every subset  $\mathbf{R} = \{\mathbf{y}_1, \dots, \mathbf{y}_k\} \subseteq \mathbf{Y}$  of size  $k$ . Let  $(\mathbf{y})_i \in Y_i$  denote the  $i^{\text{th}}$  component of the  $k$ -tuple  $\mathbf{y} \in \mathbf{Y}$ , and likewise let  $(\mathbf{b})_i \in B_i$  denote the  $i^{\text{th}}$  component of  $\mathbf{b} \in \mathbf{B}$ . Then the constraint  $C_{\mathbf{R}}$  has the following satisfaction value:

$$C_{\mathbf{R}}(\mathbf{b}(\mathbf{y}_1), \dots, \mathbf{b}(\mathbf{y}_k)) := \mathbb{E}_{\sigma \in S_k} \left[ V\left((\mathbf{y}_1)_{\sigma(1)}, \dots, (\mathbf{y}_k)_{\sigma(k)}, (\mathbf{b}(\mathbf{y}_1))_{\sigma(1)}, \dots, (\mathbf{b}(\mathbf{y}_k))_{\sigma(k)}\right) \right], \quad (128)$$

where we fix some ordering of the  $\mathbf{y}_i$ 's, like  $\mathbf{y}_1 < \cdots < \mathbf{y}_k$ . In words, we can think of  $C_{\mathbf{R}}$  as an algorithm that first randomly permutes the  $k$ -tuples  $\mathbf{y}_1, \dots, \mathbf{y}_k$  and  $\mathbf{b}(\mathbf{y}_1), \dots, \mathbf{b}(\mathbf{y}_k)$ , and that then checks “satisfaction of  $V$  along the diagonal”: i.e., does Arthur accept if, for each  $i \in [k]$ , Merlin <sub>$i$</sub>  is asked the  $i^{\text{th}}$  question in  $\mathbf{y}_i$  and responds with the  $i^{\text{th}}$  answer in  $\mathbf{b}(\mathbf{y}_i)$ ?

In this way, we ensure the following four properties:

- (1)  $\varphi$  has density  $\alpha = 1$  in the sense of Theorem 44, since it includes a constraint for every possible subset of  $k$  variables.
- (2)  $\varphi$  has alphabet size  $|\Sigma| = |\mathbf{B}| = |B_1| \cdots |B_k|$ .
- (3)  $\text{SAT}(\varphi) \geq \omega(G)$ . To see this: given any strategy  $(b_i : Y_i \rightarrow B_i)_{i \in [k]}$  for  $G$  that achieves value  $\omega$ , we can easily construct an assignment  $\mathbf{b} : \mathbf{Y} \rightarrow \mathbf{B}$  to  $\varphi$  that achieves value  $\omega$ , by setting

$$\mathbf{b}(\mathbf{y}) := (b_1((\mathbf{y})_1), \dots, b_k((\mathbf{y})_k)) \quad (129)$$

for all  $\mathbf{y} \in \mathbf{Y}$ .

- (4)  $\text{SAT}(\varphi) \leq \omega(G)$  (so in fact  $\text{SAT}(\varphi) = \omega(G)$ ). To see this: fix any assignment  $\mathbf{b} : \mathbf{Y} \rightarrow \mathbf{B}$ . Then for each  $i \in [k]$ , let  $\mathcal{D}_i$  be the probability distribution over functions  $b_i : Y_i \rightarrow B_i$  obtained by first choosing  $y_j \in Y_j$  uniformly at random for all  $j \neq i$ , and then considering the function  $b_i(y_i) := (\mathbf{b}(y_1, \dots, y_k))_i$ . Then

$$\text{SAT}(\varphi) = \mathbb{E}_{\mathbf{y}_1, \dots, \mathbf{y}_k \in \mathbf{Y}} [V((\mathbf{y}_1)_1, \dots, (\mathbf{y}_k)_k, (\mathbf{b}(\mathbf{y}_1))_1, \dots, (\mathbf{b}(\mathbf{y}_k))_k)] \quad (130)$$

$$= \mathbb{E}_{y_1 \in Y_1, \dots, y_k \in Y_k, b_1 \sim \mathcal{D}_1, \dots, b_k \sim \mathcal{D}_k} [V(y_1, \dots, y_k, b_1(y_1), \dots, b_k(y_k))] \quad (131)$$

$$\leq \omega(G), \quad (132)$$

where the last line used convexity.

Now suppose we choose a random subset  $\mathbf{I} \subseteq \mathbf{Y}$  of size

$$\kappa = \varepsilon^{-\Lambda} \log |\Sigma| = \varepsilon^{-\Lambda} \log (|B_1| \cdots |B_k|), \quad (133)$$

and consider a restriction  $\varphi_{\mathbf{I}}$  of  $\varphi$  to the subset of variables  $\{\mathbf{b}(\mathbf{y})\}_{\mathbf{y} \in \mathbf{I}}$ . Then by Theorem 44, together with properties (1) and (2) above, we have

$$\mathbb{E}_{\mathbf{I}} [\text{SAT}(\varphi_{\mathbf{I}})] \leq \text{SAT}(\varphi) + \varepsilon. \quad (134)$$

Furthermore, for each  $i \in [k]$ , let  $S_i \subseteq Y_i$  be chosen uniformly at random subject to  $|S_i| = \kappa$ , and let  $S_i = \{y_{i1}, \dots, y_{i\kappa}\}$ , fixing a uniformly-random ordering of  $y_{i1}, \dots, y_{i\kappa}$ . Also let  $S := S_1 \times \cdots \times S_k$ . Then for each  $j \in [k]$ , let  $\mathbf{y}_j := (y_{1j}, \dots, y_{kj})$ , and let  $\mathbf{I} := \{\mathbf{y}_1, \dots, \mathbf{y}_k\}$ . Then reusing the same argument from property (3) above, we have  $\omega(G_S) \leq \text{SAT}(\varphi_{\mathbf{I}})$  for every  $S$ . But the uniform distribution over  $S$ 's (and over the orderings of the elements in each  $S_i$ ) induces the uniform distribution over  $\mathbf{I}$ 's. It follows that

$$\mathbb{E}_S [\omega(G_S)] \leq \mathbb{E}_{\mathbf{I}} [\text{SAT}(\varphi_{\mathbf{I}})]. \quad (135)$$

Finally, by property (4) we have  $\text{SAT}(\varphi) \leq \omega(G)$ . Combining, we get

$$\mathbb{E}_S [\omega(G_S)] \leq \mathbb{E}_{\mathbf{I}} [\text{SAT}(\varphi_{\mathbf{I}})] \leq \text{SAT}(\varphi) + \varepsilon \leq \omega(G) + \varepsilon, \quad (136)$$

which is what we wanted to show. ■

We are now ready to prove that  $\text{AM}(k) = \text{AM}$ .

**Theorem 53**  $\text{AM}(k) = \text{AM}$  for all  $k = \text{poly}(n)$ .

**Proof.** Let  $L \in \text{AM}(k)$ . Then just like in Theorem 47, an  $\text{AM}(k)$  protocol for checking whether a string is in  $L$  can be represented as a  $k$ -player free game  $G = (Y_1, \dots, Y_k, B_1, \dots, B_k, V)$ , where  $Y_i = B_i = \{0, 1\}^{p(n)}$  for all  $i \in [k]$  and some polynomial  $p$ , and  $V$  is computable in randomized  $\text{poly}(n)$  time.

Let  $\varepsilon := \frac{1}{12}$  and  $\kappa := \varepsilon^{-\Lambda} p(n)$ . Suppose we choose  $S_i \subseteq Y_i$  uniformly at random subject to  $|S_i| = \kappa$  for all  $i \in [k]$ , then set  $S := S_1 \times \cdots \times S_k$ . Then by Theorem 52,

$$\mathbb{E}_S [\omega(G_S)] \leq \omega(G) + \frac{1}{12}. \quad (137)$$

But this immediately gives us our  $\text{AM}$  simulation, as follows. First Arthur chooses  $S_1, \dots, S_k \subseteq \{0, 1\}^{p(n)}$  uniformly at random, subject as above to  $|S_1| = \cdots = |S_k| = \kappa$ , and lets  $S = S_1 \times \cdots \times S_k$ . He then sends descriptions of  $S_1, \dots, S_k$  to Merlin, using  $k\kappa \cdot p(n) = O(k \cdot p(n)^2)$  bits. Next Merlin replies with a  $k$ -tuple of strategies  $(b_i : S_i \rightarrow B_i)_{i \in [k]}$ , which again takes  $O(k \cdot p(n)^2)$  bits. Let

$$\omega_S := \mathbb{E}_{y_1 \in S_1, \dots, y_k \in S_k} [V(y_1, \dots, y_k, b_1(y_1), \dots, b_k(y_k))] \quad (138)$$

be the subsampled success probability; notice that  $\omega_S \leq \omega(G_S)$  for all  $S$ . Then finally, Arthur computes an estimate  $\tilde{\omega}_S$  such that

$$\Pr[|\tilde{\omega}_S - \omega_S| > 0.01] \leq \exp(-\kappa) \quad (139)$$

which he can do in randomized poly( $n$ ) time, and accepts if and only if  $\tilde{\omega}_S \geq 0.51$ . (One small difference from Theorem 47 is that, even if  $V$  is deterministic, in general Arthur will still need to estimate  $\omega_S$  rather than computing it exactly. The reason is that  $\omega_S$  is an average of  $|S_1| \cdots |S_k| = \kappa^k$  terms, and  $\kappa^k$  is more than polynomial whenever  $k$  is more than constant.)

The completeness and soundness arguments are precisely the same as in Theorem 47. ■

Let us also show how, by using Theorem 52, we can go back and tighten Corollaries 50 and 51 from Section 7.3.

**Corollary 54** *Let  $G$  be a  $k$ -player free game, with question sets  $Y_1, \dots, Y_k$  and answer sets  $B_1, \dots, B_k$  (assume  $|B_i| \geq 2$  for all  $i \in [k]$ ). There exists a deterministic algorithm that approximates  $\omega(G)$  to within additive error  $\pm \varepsilon$ , in time*

$$\exp\left(\varepsilon^{-O(1)} \log(|Y_1| \cdots |Y_k|) \log(|B_1| \cdots |B_k|)\right) = n^{\varepsilon^{-O(1)} \log n}, \quad (140)$$

where  $n = |Y_1| |B_1| \cdots |Y_k| |B_k|$  is the input size.

**Proof.** Let  $\kappa := \varepsilon^{-\Lambda} \log(|B_1| \cdots |B_k|)$ . Then we simply need to loop over all possible subsets

$$S = S_1 \times \cdots \times S_k \subseteq Y_1 \times \cdots \times Y_k \quad (141)$$

such that  $|S_i| = \kappa$  for all  $i \in [k]$ . For each one, we compute the value  $\omega(G_S)$  via a brute-force search over all possible strategy  $k$ -tuples  $(b_i : S_i \rightarrow B_i)_{i \in [k]}$ . Then we output  $\tilde{\omega} := \mathbb{E}_S[\omega(G_S)]$  as our estimate for  $\omega(G)$ .

The correctness of this algorithm—i.e., the fact that  $|\tilde{\omega} - \omega| \leq \varepsilon$ —follows from Theorem 52. For the running time, note that the number of possible subsets  $S$  is

$$\binom{|Y_1|}{\kappa} \cdots \binom{|Y_k|}{\kappa} \leq (|Y_1| \cdots |Y_k|)^\kappa \leq n^{\varepsilon^{-O(1)} \log n}. \quad (142)$$

Also, for each  $S$ , the number of possible strategy  $k$ -tuples is  $|B_1|^\kappa \cdots |B_k|^\kappa \leq n^{\varepsilon^{-O(1)} \log n}$ . Hence the total running time is  $n^{\varepsilon^{-O(1)} \log n}$  as well. ■

Corollary 54, in turn, has the following further corollary.

**Corollary 55** *Assuming the Randomized ETH, any AM( $k$ ) protocol for 3SAT with a constant completeness/soundness gap must use  $\Omega(\sqrt{n})$  bits of communication in total. (Also, if Arthur's verification procedure is deterministic, then it suffices to assume the ordinary ETH.)*

**Proof.** Suppose there existed an AM( $k$ ) protocol for 3SAT, which used  $q(n) = n^{O(1)}$  bits of communication in total, and which had a completeness/soundness gap of, say,  $2/3$  versus  $1/3$  (the exact constants will be irrelevant). Set  $\varepsilon := 1/10$ . Then by Corollary 54, we can approximate the Merlins' maximum winning probability  $\omega$  to within  $\pm \varepsilon$  by a deterministic algorithm that makes  $q(n)^{\varepsilon^{-O(1)} \log q(n)} = 2^{O(\log^2 q(n))}$  evaluations of Arthur's verification function  $V$ . Furthermore, each  $V$  evaluation takes poly( $n$ ) time by a randomized algorithm if  $V$  is randomized (even counting the time needed to amplify to  $\exp(-q(n)^2)$  error probability), or poly( $n$ ) time by a deterministic algorithm if  $V$  is deterministic. Thus, the algorithm's total running time is  $2^{O(\log^2 q(n))} \text{poly}(n)$ . Moreover, the algorithm lets us decide whether  $\omega \geq 2/3$  or  $\omega \leq 1/3$ , and hence whether our original 3SAT instance was satisfiable. On the other hand, 3SAT must take  $2^{\Omega(n)}$  time assuming the ETH. Combining, we obtain  $q(n) = \Omega(\sqrt{n})$ . ■

## 8 Conclusions and Open Problems

In this paper, we saw how a deceptively simple problem—understanding the power of  $\text{AM}(2)$  protocols, and the complexity of approximating free games—hides a wealth of interesting phenomena. On the one hand, the fact that a two-prover game  $G$  is free leads to a quasipolynomial-time approximation algorithm for  $\omega(G)$ , and even a proof of  $\text{AM}(2) = \text{AM}$ . On the other hand, the fact that the Merlins still can't communicate leads to quasipolynomial-time *hardness* (assuming the ETH), and to an  $\tilde{O}(\sqrt{n})$ -communication  $\text{AM}(2)$  protocol for 3SAT.

While we managed to give nearly-matching upper and lower bounds for the complexity of  $\text{FREEGAME}$ , numerous open problems remain, both about free games themselves, and about the applicability of our techniques to other problems. We now list twelve.

- (1) Can we improve our result  $\text{NTIME}[n] \subseteq \text{AM}_{n^{1/2+o(1)}}(2)$  to  $\text{NTIME}[n] \subseteq \text{AM}_{\tilde{O}(\sqrt{n})}(2)$ ? This would follow if, for example, we could get the “best of both worlds” between the two PCP theorems of Dinur [17] and Moshkovitz and Raz [31], and achieve  $n \text{ polylog } n$  size together with a 1 vs.  $\delta$  completeness/soundness gap.
- (2) Assuming the ETH, can we completely close the gap between our  $n^{O(\varepsilon^{-2} \log n)}$  upper bound and  $n^{\tilde{\Omega}(\varepsilon^{-1} \log n)}$  lower bound on the complexity of  $\text{FREEGAME}_\varepsilon$ ? What is the right dependence on  $\varepsilon$ ? Also, given a PCP  $\phi$  of size  $N$ , is there an  $\text{AM}(2)$  protocol for verifying  $\phi$ 's satisfiability that uses  $O(\sqrt{N})$  communication rather than  $O(\sqrt{N} \log N)$ ? (In other words, in our hardness result, can we at least eliminate the log factor that comes from the birthday game, if not the log or larger factors from the PCP reduction?)
- (3) We gave two different algorithms for approximating the value of a  $k$ -player free game with  $k \geq 3$ : one that took  $n^{O(\varepsilon^{-2} k^2 \log n)}$  time (using a recursive reduction to  $(k-1)$ -player games), and one that took  $n^{\varepsilon^{-O(1)} \log n}$  time (using subsampling). Can we get the “best of both worlds,” and give an algorithm that takes  $n^{O(\varepsilon^{-2} \log n)}$  time? If so, this would imply that, assuming the ETH, any  $\text{AM}(k)$  protocol for 3SAT with a 1 vs.  $1 - \varepsilon$  completeness/soundness gap requires  $\Omega(\sqrt{\varepsilon n})$  total communication, regardless of  $k$ .
- (4) Can we prove a “Birthday Repetition Theorem” for the birthday game  $G_\phi^{k \times \ell}$ ? In other words, can we show that the Merlins' cheating probability  $\omega(G_\phi^{k \times \ell})$  continues to decrease as  $\exp(-k\ell/N)$ , if the product  $k\ell$  is larger than  $N$ ? If not, then can we give some other  $\text{AM}(k)$  protocol for 3SAT that has a 1 vs.  $\delta$  completeness/soundness gap for arbitrary  $\delta = \delta(n) > 0$ , and that uses  $n^{1/2+o(1)} \text{ polylog}(1/\delta)$  communication, rather than  $n^{1/2+o(1)} \text{ poly}(1/\delta)$ ? Directly related to that, given a free game  $G$ , can we show that deciding whether  $\omega(G) = 1$  or  $\omega(G) < \delta$  requires  $n^{\tilde{\Omega}(\frac{\log n}{\log 1/\delta})}$  time, assuming the ETH? Recall that Theorem 40 gave an  $n^{O(1 + \frac{\log n}{\log 1/\delta})}$  algorithm for that problem, while Theorem 36 gave an  $n^{\text{poly}(\delta) \cdot (\log n)^{1-o(1)}}$  lower bound assuming the ETH. Between these, we conjecture that the upper bound is tight, but the PCP and parallel-repetition machinery that currently exists seems insufficient to show this.
- (5) Given an *arbitrary* two-prover game  $G$  and positive integers  $k$  and  $\ell$ , what are the necessary and sufficient conditions on  $G, k, \ell$  for us to have  $\omega(G^{k \times \ell}) \leq \omega(G^{1 \times 1})^{\Omega(k\ell)}$ ? In other words, when exactly does birthday repetition work? Recall from Section 3.1 that, if  $\omega(G^{1 \times 1}) =$

$1 - \varepsilon$ , then we can only ever hope to do birthday repetition when  $k = O(\frac{1}{\varepsilon} \log |B|)$  and  $\ell = O(\frac{1}{\varepsilon} \log |A|)$ . Can we at least do birthday repetition up to that limit?

- (6) Can we generalize the Parallel Repetition Theorem, as well as Rao’s concentration bound (Theorem 22), to  $k$ -player free games for arbitrary  $k$ ? This would let us amplify  $\text{AM}(k)$  protocols for  $k > 2$ , though as usual with a polynomial blowup in communication cost.
- (7) Is our result that  $\text{NTIME}[n] \subseteq \text{AM}_{n^{1/2+o(1)}}(2)$ —that is, the existence of our 3SAT protocol—non-algebrizing in the sense of Aaronson and Wigderson [3]? (Recall from Proposition 38 that the result is non-relativizing.)
- (8) Can we give “direct” proofs that  $\text{AM}(k) = \text{AM}(2)$  for all  $k > 2$ , and that any  $\text{AM}(k)$  protocol can be made public-coin and perfect-completeness (where “direct” means, without using the full power of  $\text{AM}(k) = \text{AM}$ )?
- (9) How far can we improve our approximation algorithms for free games, if we assume that the game is also a *projection game* or a *unique game*? Conversely, what hardness results can we prove under those restrictions?
- (10) Let  $\text{AM}^*(2)$  be defined the same way as  $\text{AM}(2)$ , except that now the Merlins can share an unlimited amount of quantum entanglement. (Their communication with Arthur is still classical.) What can we say about this class? Does our 3SAT protocol become unsound? If so, then can we somehow “immunize” it against entangled provers—as the spectacular work of Ito and Vidick [27] (see also Vidick [36]) recently managed to do for the original BFL protocol? In the other direction, it’s currently a notorious open problem to prove *any upper bound whatsoever* on the class  $\text{MIP}^*$  (that is,  $\text{MIP}$  with entangled provers): even the set of computable languages! The issue is that we don’t have any *a priori* upper bound on the amount of entanglement the provers might need; and the more entanglement they use, the longer it could take to simulate them. Does this problem become more tractable if we restrict attention to  $\text{AM}^*$  protocols: that is, to protocols with uncorrelated questions?
- (11) Can we use our hardness result for  $\text{FREEGAME}$ —or more generally, the idea of birthday repetition—as a starting point for proving  $n^{\Omega(\log n)}$  hardness results for *other* problems? One problem of particular interest is approximate Nash equilibrium. For that problem, Lipton, Markakis, and Mehta [30] gave an  $n^{O(\varepsilon^{-2} \log n)}$  approximation algorithm—indeed, one strikingly reminiscent of our algorithm from Theorem 39—while Hazan and Krauthgamer [24] recently showed  $n^{\Omega(\log n)}$  hardness, assuming  $n^{\Omega(\log n)}$  hardness for the planted clique problem.<sup>14</sup> We conjecture that, using birthday repetition of 3SAT, one could show  $n^{\tilde{\Omega}(\varepsilon^{-1} \log n)}$  hardness for approximate Nash equilibrium assuming only the ETH. This would solve an open problem explicitly raised by Hazan and Krauthgamer.<sup>15</sup>
- (12) What can we say about  $\text{QMA}(2)$ , the class that originally motivated our study of  $\text{AM}(2)$ ? Is  $\text{QMA}(2) \subseteq \text{EXP}$ ? Are the  $\tilde{O}(\sqrt{n})$ -qubit protocols for 3SAT, due to Aaronson et al. [2] and

---

<sup>14</sup>Similarly, while this reduction is arguably weaker than the one we give, it is not hard to show that  $\text{FREEGAME}_\varepsilon$  requires  $n^{\Omega(\log n)}$  time for constant  $\varepsilon$ , under the assumption that the planted clique problem requires  $n^{\Omega(\log n)}$  time. We thank Oded Regev for this observation.

<sup>15</sup>Technically, Hazan and Krauthgamer asked for a proof that approximate Nash equilibrium is not in  $\text{P}$ , assuming  $\text{MAX CLIQUE}$  requires  $2^{\omega(\sqrt{n})}$  time. But this is extremely similar to assuming the ETH.

Harrow and Montanaro [23], optimal assuming the ETH? Is the  $\text{BSS}_\epsilon$  problem from Section 4 solvable in  $n^{O(\epsilon^{-2} \log n)}$  time, as  $\text{FREEGAME}_\epsilon$  is?

## 9 Acknowledgments

We thank Boaz Barak, Oded Regev, Avi Wigderson, and other participants at the 2013 Banff Complexity Theory workshop for helpful discussions. We especially thank Peter Shor for early discussions, Ryan O’Donnell for pointing us to [5] and [11], Anup Rao for clarifications about parallel repetition, and Aram Harrow for goading us to write this paper up after a four-year delay.

## References

- [1] S. Aaronson and A. Ambainis. The need for structure in quantum speedups. In *Proc. Innovations in Theoretical Computer Science (ITCS)*, 2011. arXiv:0911.0996.
- [2] S. Aaronson, S. Beigi, A. Drucker, B. Fefferman, and P. Shor. The power of unentanglement. In *Proc. IEEE Conference on Computational Complexity*, pages 223–236, 2008. arXiv:0804.0802.
- [3] S. Aaronson and A. Wigderson. Algebrization: a new barrier in complexity theory. *ACM Trans. on Computation Theory*, 1(1), 2009. Conference version in Proc. ACM STOC 2008.
- [4] D. Aharonov and T. Naveh. Quantum NP - a survey. quant-ph/0210077, 2002.
- [5] N. Alon, W. F. de la Vega, R. Kannan, and M. Karpinski. Random sampling and approximation of MAX-CSPs. *J. Comput. Sys. Sci.*, 67(2):212–243, 2003. Earlier version in STOC’2002.
- [6] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *J. ACM*, 45(3):501–555, 1998.
- [7] S. Arora and S. Safra. Probabilistic checking of proofs: a new characterization of NP. *J. ACM*, 45(1):70–122, 1998.
- [8] L. Babai, L. Fortnow, and C. Lund. Nondeterministic exponential time has two-prover interactive protocols. *Computational Complexity*, 1(1):3–40, 1991.
- [9] L. Babai and S. Moran. Arthur-Merlin games: a randomized proof system, and a hierarchy of complexity classes. *J. Comput. Sys. Sci.*, 36:254–276, 1988.
- [10] B. Barak, F. Brandão, A. Harrow, J. Kelner, D. Steurer, and Y. Zhou. Hypercontractivity, sum-of-squares proofs, and their applications. In *Proc. ACM STOC*, pages 307–326, 2012. arXiv:1205.4484.
- [11] B. Barak, M. Hardt, T. Holenstein, and D. Steurer. Subsampling mathematical programs and average-case complexity. In *Proc. ACM-SIAM Symp. on Discrete Algorithms (SODA)*, pages 512–531, 2011.
- [12] B. Barak, A. Rao, R. Raz, R. Rosen, and R. Shaltiel. Strong parallel repetition theorem for free projection games. In *Proc. APPROX-RANDOM*, pages 352–365, 2009.

- [13] H. Blier and A. Tapp. All languages in NP have very short quantum proofs. arXiv:0709.0738, 2007.
- [14] F. Brandão, M. Christandl, and J. Yard. A quasipolynomial-time algorithm for the quantum separability problem. In *Proc. ACM STOC*, pages 343–351, 2011. arXiv:1011.2751.
- [15] F. Brandão and A. Harrow. Quantum de Finetti theorems under local measurements with applications. In *Proc. ACM STOC*, pages 861–870, 2013. arXiv:1210.6367.
- [16] J. Chen and A. Drucker. Short multi-prover quantum proofs for SAT without entangled measurements. arXiv:1011.0716, 2010.
- [17] I. Dinur. The PCP theorem by gap amplification. *J. ACM*, 54(3):12, 2007.
- [18] U. Feige and O. Verbitsky. Error reduction by parallel repetition - a negative result. *Combinatorica*, 22(4):461–478, 2002.
- [19] L. Fortnow and M. Sipser. Are there interactive protocols for co-NP languages? *Inform. Proc. Lett.*, 28:249–251, 1988.
- [20] M. Furer, O. Goldreich, Y. Mansour, M. Sipser, and S. Zachos. On completeness and soundness in interactive proof systems. In *Advances in Computing Research: A Research Annual*, volume 5, pages 429–442. 1989.
- [21] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *J. ACM*, 45(4):653–750, 1998.
- [22] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof systems. In *Randomness and Computation*, volume 5 of *Advances in Computing Research*. JAI Press, 1989.
- [23] A. Harrow and A. Montanaro. Testing product states, quantum Merlin-Arthur games and tensor optimisation. *J. ACM*, 60(1), 2013. arXiv:1001.0017. Earlier version in Proceedings of IEEE FOCS’2010.
- [24] E. Hazan and R. Krauthgamer. How hard is it to approximate the best Nash equilibrium? *SIAM J. Comput.*, 40(1):79–91, 2011. Previous version in SODA’2009.
- [25] T. Holenstein. Parallel repetition: simplification and the no-signaling case. *Theory of Computing*, 5(1):141–172, 2009. arXiv:cs/0607139.
- [26] R. Impagliazzo and R. Paturi. Complexity of k-SAT. In *Proc. IEEE Conference on Computational Complexity*, pages 237–240, 1999.
- [27] T. Ito and T. Vidick. A multi-prover interactive proof for NEXP sound against entangled provers. In *Proc. IEEE FOCS*, pages 243–252, 2012. arXiv:1207.0550.
- [28] H. Kobayashi, K. Matsumoto, and T. Yamakami. Quantum Merlin-Arthur proof systems: are multiple Merlins more helpful to Arthur? In *ISAAC*, pages 189–198, 2003. quant-ph/0306051.

- [29] J. Komlós and M. Simonovits. Szemerédi’s regularity lemma and its applications in graph theory. In *Combinatorics, Paul Erdős is eighty*, volume 2, pages 295–352. János Bolyai Math. Soc., 1996.
- [30] R. J. Lipton, E. Markakis, and A. Mehta. Playing large games using simple strategies. In *ACM Conference on Electronic Commerce*, pages 36–41, 2003.
- [31] D. Moshkovitz and R. Raz. Two-query PCP with subconstant error. *J. ACM*, 57(5), 2010. Previous version in FOCS’2008. TR08-071.
- [32] A. Rao. Parallel repetition in projection games and a concentration bound. *SIAM J. Comput.*, 40(6):1871–1891, 2011. ECCC TR08-013. Earlier version in STOC’2008.
- [33] R. Raz. A parallel repetition theorem. *SIAM J. Comput.*, 27(3):763–803, 1998. Earlier version in STOC 1995.
- [34] R. Shaltiel. Derandomized parallel repetition theorems for free games. *Computational Complexity*, 22(3):565–594, 2013. Earlier version in CCC’2010.
- [35] I. Tourlakis. Time-space tradeoffs for SAT on nonuniform machines. *J. Comput. Sys. Sci.*, 63(2):268–287, 2001.
- [36] T. Vidick. Three-player entangled XOR games are NP-hard to approximate. In *Proc. IEEE FOCS*, 2013. arXiv:1302.1242.